The Islamic University of Gaza
Deanery of Graduate Studies
Faculty of Engineering
Electrical Engineering Department

Model Order Reduction Using LMI

By

Mohammed S. R. Abu Hatab

Supervisor

Dr. Hatem Elaydi

"A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Science in Electrical Engineering."

1430 - 2009

بسم الله الرحمن الرحيم

الجامعة الإسلامية – غزة
The Islamic University - Gaza

عمادة الدراسات العليا
هاتف داخلي: 1150

الرقم Ref. ج س غ/35/
التاريخ Date 2009/05/24

نتيجة الحكم على أطروحة ماجستير

بناءً على موافقة عمادة الدراسات العليا بالجامعة الإسلامية بغزة على تشكيل لجنة الحكم على أطروحة الباحث/**محمد سليمان رزق أبو حطب** لنيل درجة الماجستير في كلية *الهندسة* – قسم **الهندسة الكهربائية/أنظمة تحكم** وموضوعها:

"Model Order Reduction Using LMI"

وبعد المناقشة العلنية التي تمت اليوم الاثنين 07 جمادى آخر 1430هـ، الموافق 2009/06/01م الساعة الثانية عشرة ظهراً، اجتمعت لجنة الحكم على الأطروحة والمكونة من:

د. حاتم العايدي	مشرفاً ورئيساً	.....
د. باسل حمد	مناقشاً داخلياً	.....
د. محمد حسين	مناقشاً داخلياً	.....

وبعد المداولة أوصت اللجنة بمنح الباحث درجة الماجستير في كلية *الهندسة*/قسم **الهندسة الكهربائية/أنظمة تحكم**.

واللجنة إذ تمنحه هذه الدرجة فإنها توصيه بتقوى الله ولزوم طاعته وأن يسخر علمه في خدمة دينه ووطنه.

والله ولي التوفيق،،،

عميد الدراسات العليا

د. زياد إبراهيم مقداد

P.O. Box 108, Rimal, Gaza, Palestine fax: +970 (8) 286 0800 فاكس tel: +970 (8) 286 0700 تلفون ص.ب 108 الرمال، غزة، فلسطين
public@iugaza.edu.ps www.iugaza.edu.ps

Abstract

In this thesis, the problem of Frobenius Hankel (FH) norm, H_2 norm and H_∞ norm reduced order approximations will be investigated. The necessary and sufficient conditions for the existence of an approximate solution within a specified error γ will be found, these conditions are given in terms of a set of linear matrix inequalities (LMI) and a matrix rank constraint for both continuous and discrete time multi input-multi output systems. The alternating projection algorithm (APA) and the cone complementarity algorithm (CCA) are used to solve the rank constraint problem and a comparison between both algorithms is presented. Numerical algorithms which use the cone complementary algorithm and alternating projection method are proposed and a method of finding an initial starting point is suggested. Comparison between H_2, H_∞ and FH norms model reduction using $LMI's$ techniques is discussed to show the effectiveness of these methods. The proposed reduction method is extended to polytopic uncertain systems to show the effectiveness of model order reduction using LMIs. Numerical examples are given to show and validate the effectiveness of the FH norm and H_∞ norm reduced order approximations using cone complementary algorithm to find at least as good approximates as other methods.

ملخص الرســـالة:

تهدف هذه الدراسة إلى معرفة فاعلية استخدام المصفوفات الخطية غير المتساوية في تصغير درجة النموذج و لتحقيق هذا الهدف تسعى الدراسة إلى الإجابة عن الأسئلة الآتية:

1- ما النموذج المقترح لتصغير درجة النموذج الحقيقي؟

2- ما فاعلية النموذج المقترح في تصغير درجة النموذج الحقيقي؟

و لهذا الغرض اعد الباحث نموذجا قائما علي المصفوفات الخطية غير المتساوية معتمدةً علي كـــل من مقياس فوربينيس هانكل و المقياس الثنائي و المقياس اللانهائي ولتطبيق هـــذه المقـــاييس علـــي المصفوفات الخطية غير المتساوية وجد الباحث الشروط اللازمة والكافية لوجود نمـــوذج مقـــارب للنموذج الحقيقي بدرجة أقل بحيث تكون نسبة الخطأ بين النموذجين قيمة صغيرة محددة عبارة عن عدة مصفوفات خطية غير متساوية وقيد علي رتبة مصفوفة بحيث تشمل النتائج الأنظمة المتـــصلة والمنقطعة ذات المداخل والمخارج المتعددة.

ولتطبيق هذه المقاييس علي المصفوفات الخطية غير المتساوية استخدم الباحث طريقتي الإسقاط التتابعية والمخروط المكملة لفك قيد رتبة المصفوفة وتحويلها إلى مصفوفات خطية غير متساوية ثم قام الباحث بالإجرءات الآتية :

أ. تحديد نقطة بداية أساسية لانطلاق النموذج المقترح.

ب. فك قيد رتبة المصفوفة بحيث يمكن التعامل معها.

ت. تطبيق المقاييس الثلاث السابقة في تنفيذ النموذج المقترح.

وباستخدام النموذج المقترح توصل الباحث إلى نتائج أفضل من النتائج التي تم التوصل إليها سابقاً بطرق مختلفة مما يؤكد فاعلية النموذج المقترح في تصغير درجة النموذج الحقيقي كما قام الباحث بالمقارنة بين المقاييس السابقة بطريقتي الإسقاط التتابعية والمخروط المكملة وقام بتطوير هذا النموذج باستخدام طريقة المخروط المكملة بحيث يمكن استخدامه في تصغير درجة نموذج غير ثابت تكون فيه بعض عناصر النموذج عبارة عن فترات خطية وفي ضوء هذه النتائج وجد الباحث أن طريقة المخروط المكملة ذات فاعلية أكثر من طريقة الإسقاط التتابعية كما وجد أن تصغير درجة النموذج باستخدام مقياس فوربينيس هانكل يعطي نتائج أفضل من المقياسين الآخرين.

يوصي الباحث باعتماد هذا النموذج لتصغير درجة النموذج الحقيقي لعمل المتحكمات اللازمة لأي نموذج و ذلك لفاعليته في تصغير درجة النموذج بنسبة خطأ صغيرة جداً.

Dedication

For the soul of my father, my mother, my wife and my sweet kids Sulaiman, Abdelrahman, Janat and Dania

Acknowledgements

First and foremost, all praise is due to Allah, the Almighty, who gave me the opportunity, strength, and patience to carry out this work.

I would like to give my sincere gratitude to my supervisor Assistant Prof. Dr. Hatem Elaydi for his continuous support, great guidance, endless help, good knowledge and huge confidence he gave me.

Special thanks go to Assistant Prof. Dr. Basiel Hamed and Associate Prof. Dr. Mohammed T. Hussein, thesis examinors, for their patient guidance and generous support for this research. Many thanks to my department and fellow associates for their help and support during my course of study. Finally, words will not be enough to thank my family for their infinite support and patience.

Contents

List of Tables

List of Figures

Chapter 1

Introduction and Literature Review

In this chapter, we will introduce some of the popular methods to reduce the complexity of models, which depends mainly on the balanced state space representation and the Hankel singular values. These methods are balanced truncation and Hankel norm reduction methods. Although, these methods generally do not give optimal or suboptimal reduced order system, we can still use it as our starting points on LMI based model reduction schemes.

1.1 Introduction

Simple models are preferred above complex models and accurate models are preferred above inaccurate models. To obtain high accuracy models, we usually need to implement complex models, while simple models are generally inaccurate. In this chapter, we assume that a *stable linear time-invariant system* is given and we address the problem to approximate this system by a simpler one. The approximate system is required to have a dynamic behavior which is as close as possible, to the behavior of the system which we wish to approximate. The problem on this thesis is an optimal model approximation. This problem is definitely a relevant one as many models derived from first principles or identification routines tend to become complex. Also, in the design and synthesis of control systems, controllers may become too complex to be implemented.

The complexity of linear time-invariant models is generally defined as the dimension of the state vector of any minimal state space representation of the system. This number is also known as the McMillan degree or the order of the system. After the definition of complexity the model approximation problem can be stated as follows:
Given a stable, linear time-invariant system $G(s)$ of McMillan degree n, find a lower order linear, time-invariant system $\hat{G}(s)$ such that the behavior B of $G(s)$ is close to the behavior $\hat{B}$ of $\hat{G}(s)$.

There are a large number of techniques available for deriving reduced order models and lower order controllers. One of the most commonly used methods is the balanced truncation method. The procedure is easy to be implemented and

also the method is extensively studied [1]. Another method is the Hankel norm approximation [2]. As we can recognize from the model reduction techniques, there is an error between the original high order system and the obtained reduced order model in some sense as an index of how good the approximate is. For both of the methods upper bounds on the error in the H_∞ sense and also a lower bound for the Hankel norm approximation method are expressed in terms of the Hankel singular values of the original system. The previous methods do not in general produce optimal approximates in the H_∞ sense and there are several methods for H_∞ optimal model reduction are developed to reduce the error γ between the reduced and the original model [3, 4] are examples of the developed H_∞ optimal model reduction.

1.2 State Truncations

Consider a dynamical system in input-state-output form:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned} \tag{1.1}$$

Here, we have a system with n states, m inputs and p outputs. That is, $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ and $y(t) \in \mathbb{R}^p$ for all time instants $t \in \mathbb{R}$. Suppose that the states x of this system are partitioned in two components as:

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

Any such partitioning causes a compatible partitioning of the system matrices as follows:

$$\mathbf{A} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \qquad \mathbf{B} = \begin{pmatrix} B_1 \\ B_2 \end{pmatrix}, \qquad \mathbf{C} = \begin{pmatrix} C_1 & C_2 \end{pmatrix}$$

The matrix D is not affected by the partitioning of the state. If we assume that the vector x_1 has dimension k, then $A_{11} \in \mathbb{R}^{k\times k}, B1 \in \mathbb{R}^{k\times m}, C1 \in \mathbb{R}^{p\times k}\, and\, D \in \mathbb{R}^{p\times m}$. We will have (A_{11}, B_1, C_1, D) as a k^{th} order truncation of (A, B, C, D). This k^{th} order truncation for the system (1.1) is:

$$\begin{aligned} \dot{\xi}(t) &= A_{11}\xi(t) + B_1 u(t) \\ y(t) &= C_1\xi(t) + Du(t) \end{aligned} \tag{1.2}$$

Although, the original system is stable, controllable and minimal the truncated system may not be.

1.3 Modal Truncations

Consider a state space transformation:

$$x = T\acute{x} \tag{1.3}$$

for the system (1.1) with T a non-singular matrix of dimension $n \times n$. Since such a transformation only amounts to rewriting the state variable in a new basis, this transformation does not affect the input-output behavior associated with (1.1). Thus,

Theorem 1.1. *The If Σ is represented by* (1.1)*, then the external (or input-output behavior) of Σ is equivalently represented by the input-state-output model*

$$\begin{aligned} \dot{\acute{x}}(t) &= T^{-1}AT\acute{x}(t) + T^{-1}Bu(t) \\ y(t) &= CT\acute{x}(t) + Du(t) \end{aligned} \tag{1.4}$$

Proof. Can be obtained from (1.1) by substituting (1.3) in (1.1) and solving for $\acute{x}$ □

In fact, we describe all minimal input-state-output representations of Σ by varying T over the set of non-singular matrices.
The transformation:

$$A \rightarrow T^{-1}AT = \acute{A}$$

is called *a similarity transformation* of the matrix A. The *characteristic polynomial* of the A matrix occurring in (1.1) is the polynomial $p(s) = det(sI - A)$.
We can write this polynomial in various formats

$$\begin{aligned} p(s) &= det(sI - A) \\ &= p_0 + p_1 s + ... + p_n s^n \\ &= (s - \lambda_1)(s - \lambda_2)...(s - \lambda_n) \end{aligned} \tag{1.5}$$

where $\lambda_1, ..., \lambda_n$ are the so called modes of the system. For the modal canonical form we assume that the natural frequencies $\lambda_1, \lambda_2, ...\lambda_n$ are all different. For each natural frequency λ_i there exists a (complex) eigenvector v_i of dimension n such that $[\lambda_i - A]v_i = 0$. If we store these eigenvectors $v_1, ..., v_n$ in one $n \times n$ matrix $T = [v_1\, v_2 \,...\, v_n]$ then we obtain a non-singular transformation (1.3) and the transformed A matrix takes the form

$$\acute{A} := T^{-1}AT = diag(\lambda_1\,, ...,\, \lambda_n) \tag{1.6}$$

which is called a Jordan form of the matrix A. The resulting state space system is said to be in *modal canonical form*:

Definition 1.1:(Modal canonical form) *The input-state-output system:*

$$\begin{aligned} \dot{x}(t) &= \acute{A}x(t) + \acute{B}u(t) \\ y(t) &= \acute{C}x(t) + \acute{D}u(t) \end{aligned} \tag{1.7}$$

with $\acute{A} = T^{-1}AT$ *as in* (1.6), $\acute{B} = T^{-1}B$, $\acute{C} = TC$ *and* $\acute{D} = D$ *is called a* ***modal canonical state space representation****.*

Now, suppose that the system (1.1) is stable. This implies that the modes or the eigenvalues of the system has a negative real part. And we have $0 < |\lambda_1| < |\lambda_2| < ... < |\lambda_n|$. If we partition

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

where x_1 has dimension $k < n$, then the truncated system is defined by leaving out the fast modes of the system.

Definition 1.2:(Modal truncations) *If* (1.7) *is a modal canonical state space system, then the* k^{th} *order truncation*

$$\begin{aligned} \dot{\xi}(t) &= \acute{A_{11}}\xi(t) + \acute{B_1}u(t) \\ y(t) &= \acute{C_1}\xi(t) + \acute{D}u(t) \end{aligned} \tag{1.8}$$

is called the k^{th} *order modal truncation of* (1.1).

1.4 Balanced Truncations

A second popular procedure for model approximation is the method of balanced truncations. It requires a state truncation of a system which is represented in balanced state space form. The balanced state space representation is an input-state-output representation of the form (1.1) for which the controllability grammian and the observability grammian are equal and diagonal.

1.4.1 Balanced state space representations

Suppose that a minimal and stable state space representation (1.1) of a dynamical system is given. We define two matrices.
The controllability grammian associated with the system (A, B, C, D) is the matrix

$$P \triangleq \int_0^\infty e^{At}BB^T e^{A^T t}dt \tag{1.9}$$

Since the system is assumed to be stable, the eigenvalues of A has a negative real part, and from this it follows that the integral in (1.9) is well defined. Note that P is an $n \times n$ real matrix, it is symmetric.
The observability grammian associated with the system (A, B, C, D) is the matrix

$$Q \triangleq \int_0^\infty e^{A^T t} C^T C e^{At} dt \tag{1.10}$$

Again, the stability assumption implies that the integral in (1.10) is well defined. Q is an $n \times n$ real symmetric matrix.
Fortunately, to compute the controllability and observability grammians of a state space system, it is not necessary to perform the integration as in (1.9) and (1.10) the next theorem tell us how to obtain the grammians from the Lyapunov equation.

Theorem 1.2. *Given a minimal and stable system* (1.1)*, its controllability grammian P is the unique positive definite solution of the Lyapunov equation*

$$AP + PA^T + BB^T = 0. \tag{1.11}$$

Similarly, the observability grammian Q is the unique positive definite solution of

$$A^T Q + QA + C^T C = 0. \tag{1.12}$$

If the system we have is minimal, then grammians P and Q are the unique solutions to (1.11) and (1.12), respectively. The computation of the grammians is therefore equivalent to the algebraic problem to find solutions of Lyapunov equations (1.11) and (1.12). Balanced state space representations are now defined as follows.

Definition 1.3: *A minimal state space representation* (1.1) *is called balanced if the controllability and observability grammians are equal and diagonal, i.e, if*

$$P = Q = diag(\sigma_1, \sigma_2, ...\sigma_n)$$

where σ_i *are real and positive numbers that are ordered according to*

$$\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_n > 0.$$

1.4.2 Existence of Balanced State Space Representations

To find the balanced representation of the system (1.1), let us assume that we calculated the controllability and observability grammians for the stable system (1.1) and let us see how these grammians transform if we change the basis of the state space. Thus, consider again the state space transformation (1.3). As we have seen, this results in the transformed state space parameters $(\acute{A}, \acute{B}, \acute{C}, \acute{D})$ as shown on definition(1.1) yields that the transformed grammians take the form

$$\acute{P} = T^{-1}P(T^{-1})^T; \qquad \acute{Q} = T^T Q T.$$

This shows that the grammians depend strongly on the basis of the state space. However, their product

$$\acute{P}\acute{Q} = T^{-1}P(T^{-1})^T T^T Q T = T^{-1}PQT$$

so that The eigenvalues of PQ are invariant under state space transformations.
Let $\lambda_1, ..., \lambda_n$ denote the eigenvalues of the product PQ. Then λ_i are positive real numbers for $i = 1, ..., n$ so that it makes sense to consider their square roots

$$\sigma_i := \sqrt{\lambda_i} = \lambda_i^{1/2}(PQ).$$

We just showed that these numbers are system invariants: they do not change by transforming the basis of the state space. In the literature, these system invariants play a crucial role and are called the *Hankel singular values of the system* (1.1) [2].
To show that balanced state space representations actually exist, we need to construct a non-singular state transformation matrix T that simultaneously diagonalizes the controllability and the observability grammians P and Q.
The algorithm (which is of course implemented in MATLAB) is as follows

INPUT: State space parameters (A, B, C, D) of a minimal, stable system of the form (1.1)

Step 1: Compute the grammians P and Q.

Step 2: Factorize $P = R^\top$ (the routine chol in MATLAB is doing this for you)

Step 3: Construct the matrix $RQR^\top$ and (since it is positive definite) factorize it as $RQR^\top = U^\top \Sigma^2 U$ where U is a unitary matrix (i.e., $UU^\top = U^\top U = I$) and $\Sigma = diag(\sigma_1, \sigma_2, ..., \sigma_n)$ then the numbers σ_i are the Hankel singular values (i.e., the square roots of the eigenvalues $\lambda_i(PQ)$).

Step 4: Define the non-singular matrix $T := R^\top U \Sigma^{-1/2}$.

OUTPUT: the matrices $(\acute{A}, \acute{B}, \acute{C}, \acute{D})$ as defined in definition(1.1).

It is easily seen that the state transformation defined in step 4 of the algorithm achieves that the grammians of the transformed system are

$$\acute{Q} = T^\top QT = \Sigma^{-1/2}U^\top RQR^\top U\Sigma^{-1/2} = \Sigma^{-1/2}\Sigma^2\Sigma^{-1/2} = \Sigma$$
$$\acute{P} = T^{-1}P(T^{-1})^{-\top} = \Sigma^{1/2}U^\top R^\top PR^{-1}U\Sigma^{1/2} = \Sigma^{1/2}\Sigma^{1/2} = \Sigma$$

i.e., they are equal and diagonal with the Hankel singular values as diagonal elements. We thus proved the following important result

Theorem 1.3. *Every stable dynamical system of the form* (1.1) *admits a balanced input-state-output representation.*

1.4.3 Balanced Truncations

The above interpretation justifies the following definition of a model reduction procedure based on balanced state space representations. Suppose we decompose the state variable x of a balanced state space system as

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}.$$

where x_1 has dimension k. Then x_1 can be regarded as the k most important state components in view of both their controllability and observability properties. A state truncation of this type is called a balanced state truncation

Definition 1.4(Balanced Truncations): *If* (1.1) *is a stable, balanced state space system, then the* k^{th} *order truncation*

$$\begin{aligned} \dot{\xi}(t) &= A_{11}\xi(t) + B_1u(t) \\ y(t) &= C_1\xi(t) + Du(t) \end{aligned} \tag{1.13}$$

is called the k^{th} *balanced truncation of* (1.1).

This simple approximation method provides very efficient and good approximate models. It eliminates the poorly controllable and poorly observable states from a state space model. The number k may in practice be determined by inspecting the ordered sequence of Hankel singular values $\sigma_1, ..., \sigma_n$. A drop in this sequence (i.e., a number k for which $\sigma_{k+1}/\sigma_k << 1$) may give you a reasonable estimate of the order of a feasible approximate model. If $\sigma_k > \sigma_{k+1}$ (as will be the case in many practical situations) the k^{th} order balanced truncation turns out to have good properties:

Theorem 1.4. *Suppose that* (1.1) *is a balanced state space representation of a stable system. Let* $k < n$ *and suppose that* $\sigma_k > \sigma_{k+1}$. *Then the* k^{th} *order balanced truncation is minimal, stable, balanced.*

Now let us consider the following remarks:

Remark 1.1: If $G(s)$ denotes the transfer function corresponding to (1.1) and $G_k(s)$ is the transfer function of a k^{th} order balanced truncation of $G(s)$ then it is known that the error $G - G_k$ satisfies

$$\|G(s) - G_k(s)\|_\infty \leq 2(\sigma_{k+1} + \sigma_{k+2} + ... + \sigma_n). \tag{1.14}$$

Thus the maximum peak in the Bode diagram of the error system is less than twice the sum of the tail of the Hankel singular values.

Remark 1.2: All the results of this section can be repeated for **discrete time systems**. Formulas change, but the ideas are identical.

Remark 1.3: In MATLAB the relevant routines for constructing balanced state space models are balreal for *continuous time systems* and dbalreal for *discrete time systems*.

1.5 Hankel Norm Reductions

The Hankel norm reductions are among the most important techniques of model reduction procedures that exist today. It is one of the model approximation procedures that produce optimal approximate models according to some well-defined criterion that we will introduce below. It constitutes a beautiful theory associated with the names of Nehari, Arov- Adamjan-Krein (AAK) and Glover [2, 5]. Glover introduced state space ideas in this problem area and in our exposition we will follow his work.

1.5.1 Hankel Singular Values and the Hankel Norm

The Hankel norm of a system is easily computed. In fact, it turns out to be equal to the maximal Hankel singular value for the systems. For Discrete time systems is straightforward:
The controllability grammian is the positive definite matrix

$$P \triangleq \sum_{k=0}^{\infty} A^k B B^T (A^T)^k \tag{1.15}$$

The observability grammian is the matrix

$$Q \triangleq \sum_{k=0}^{\infty} (A^T)^k C^T C A^k \tag{1.16}$$

The grammians of the system are the unique positive definite solution to the Lyapunov equations

$$APA^T + BB^T - P = 0. \tag{1.17}$$

$$A^TQA + C^TC - Q = 0. \tag{1.18}$$

These equations form an efficient approach to solve for the grammians.

Definition 1.5: *The **Hankel singular values** of $G(s) \in H_2$ are given by*

$$\sigma_i(G(s)) \triangleq [\lambda_i(PQ)]^{\frac{1}{2}} \tag{1.19}$$

Where P and Q are the controllability and observability grammians of $G(s)$.

As in the previous section, the eigenvalues $\lambda_1, ..., \lambda_n$ of the product PQ are input-output invariants and their square roots $\sigma_1, ..., \sigma_n$ are called the Hankel singular values. We assume that the Hankel singular values are ordered according to $\sigma_1 \geq \sigma_2 \geq ..... \geq \sigma_n \geq 0$ and we obtain the followingt result .

Theorem 1.5. *If the system Σ is stable and represented by* (1.1)*, then The Hankel norm*

$$\|\Sigma\|_H = \lambda_{max}^{1/2}(PQ) = \sigma_1.$$

Proof. The proof of this theorem cab be found on [2]. □

Thus the Hankel norm is nothing else than the largest Hankel singular value of the system and it can be computed directly from the product of the two grammians associated with a state space representation of the system. The same result holds for continuous and discrete time systems.

1.5.2 The Hankel Norm Model Reduction Problem

In the previous section we have seen how a balanced representation can lead to a reduced order model. However, this algorithm did not allow for an interpretation as an optimal approximation. That is, the model obtained by balanced truncation did not minimize a criterion in which we agreed how far the n^{th} order system Σ is apart from a k^{th} order approximation Σ_k. The Hankel-norm model reduction problem does involve such a criterion.
Given an n^{th} order stable system Σ, find a k^{th} order stable system Σ_k so as to minimize the Hankel norm of the error $\|\Sigma - \Sigma_k\|_H$.

Here, $\Sigma - \Sigma_k$ is the error system which has the difference of the outputs of Σ and Σ_k as its output. It turns out that the optimal model approximation problem is solvable in the Hankel norm! Even though the Hankel norm does not allow a very natural system theoretic interpretation this is a remarkable result. The main result of this section is given in the algorithm below and provides an entire algorithm to obtain a state space representation (A_k, B_k, C_k, D_k) of the optimal Hankel-norm approximant Σ_k of a stable system Σ in state space form [2].

INPUT The system (A, B, C, D) with (A, B) controllable (C, A) observable and A stable.

DESIRED A system (A_k, B_k, C_k, D_k) of order $\leq k$ which approximates the system (A, B, C, D) optimal in the Hankel norm.

Algorithm:

Step 1: Compute the Hankel singular values $\sigma_1, ..., \sigma_n$ of (A, B, C, D) and assume that

$$\sigma_k > \sigma_{k+1} = \sigma_{k+2} = ... = \sigma_{k+r} > \sigma_{k+r+1} \geq ... \geq \sigma_n > 0$$

i.e., σ_{k+r} has multiplicity r.

Step 2: Transform (A, B, C, D) to a partially balanced form

$$\mathbf{P} = \mathbf{Q} = \begin{pmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{pmatrix}.$$

where $\Sigma_1 = diag(\sigma_1, ..., \sigma_k, \sigma_{k+r+1}, ..., \sigma_n)$ and $\Sigma_2 = \sigma_{k+1} I_r$. That is, the $(k+1)^{st}$ Hankel singular value is put in the south-east corner of the joint gramians.

Step 3: Partition (A, B, C, D) conformally with the partitioned gramians as

$$\mathbf{A} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \qquad \mathbf{B} = \begin{pmatrix} B_1 \\ B_2 \end{pmatrix}, \qquad \mathbf{C} = \begin{pmatrix} C_1 & C_2 \end{pmatrix}$$

Further define

$$\Gamma = \Sigma_1^2 - \sigma_{k+1}^2 I.$$

and note that Γ is non-singular.
If $m \leq p$, proceed. If $m > p$, replace (A, B, C, D) by (A^T, C^T, B^T, D^T) and proceed.

Step 4: Determine a unitary matrix U satisfying $B_2 + C_2^T U = 0$.

Step 5: Let $\hat{n} := n - r$ be the state space dimension of the system defined as

$$\begin{aligned} \hat{A} &= \Gamma^{-1}(\sigma_{k+1}^2 A_{11}^\top + \Sigma_1 A_{11} \Sigma_1 - \sigma_{k+1} C_1^\top U B_1^\top) \\ \hat{B} &= \Gamma^{-1}(\Sigma_1 B_1 + \sigma_{k+1} C_1^\top U) \\ \hat{C} &= C_1 \Sigma_1 + \sigma_{k+1} U B_1^\top \\ \hat{D} &= D - \sigma_{k+1} U \end{aligned}$$

The gain of the error transfer function is

$$\sigma_{max}(T(j\omega) - \hat{T}(j\omega)) = \sigma_{k+1}$$

for all $\omega \in \mathbb{R}$. The system $\hat{\Sigma}$ is in general not stable, though.

Step 6: Determine the stable subsystem of $\hat{\Sigma}$ by choosing a basis of the state space of $\hat{\Sigma}$ such that

$$\hat{A} = \begin{pmatrix} \hat{A}_- & 0 \\ 0 & \hat{A}_+ \end{pmatrix}, \qquad \hat{B} = \begin{pmatrix} \hat{B}_- \\ \hat{B}_+ \end{pmatrix}, \qquad \hat{C} = \begin{pmatrix} \hat{C}_- & \hat{C}_+ \end{pmatrix}$$

where A_- has all its eigenvalues in the open left half complex plane and A_+ all its eigenvalues in the open right half complex plane, i.e., A_- is stable and A_+ is anti-stable. $\hat{A}$ will not have eigenvalues on the imaginary axis while $\hat{A}_-$ will have dimension $\leq k$.
If $m \leq p$, proceed. If $m > p$, replace $(\hat{A}, \hat{B}, \hat{C}, \hat{D})$ by $(\hat{A}^\top, \hat{B}^\top, \hat{C}^\top, \hat{D}^\top)$ and proceed.

OUTPUT: Set

$$\begin{aligned} A_k &= \hat{A}_- \\ B_k &= \hat{B}_- \\ C_k &= \hat{C}_- \\ D_k &= \hat{D}_- \end{aligned}$$

Then the system Σ_k defined by

$$\begin{aligned} \frac{d\xi}{dt} &= A_k\xi(t) + B_k u(t) \\ y(t) &= C_k\xi(t) + D_k u(t) \end{aligned}$$

is a state space representation of an optimal Hankel norm approximant of Σ and the error

$$\|\Sigma - \Sigma_k\|_H = \sigma_{k+1}.$$

There are number of approaches, such as [6, 7, 8, 9], use first order necessary conditions for optimality and develop optimization algorithms to find solutions to resulting nonlinear equations. Most of the methods in this direction are only applicable to the single input single output (SISO) case. Furthermore, it can be recognized from [10, 11] that whether the global optimum is always achievable is unclear in the continuous time case (while it is shown to exist in the discrete time case [12]) and that, in the case of nonexistence of the optimum, these approaches can only find local optima which may be far from the true (global) optimum. Even if the existence of the global optimum is guaranteed, optimization methods based on search

algorithms can have difficulties [13]. There may be one or more local optima and it is difficult to guarantee that the obtained solution is close to the global optimum. Moreover, there is usually no guarantee that the chosen stopping criterion for such a search algorithm is appropriate. To overcome these problems, several algorithms based on algebraic methods have been proposed that directly solve a set of nonlinear equations [13, 14, 15]. These approaches seem to have potential (in cases where the optima are achievable), but computation cost required for such approaches is still high and structural properties of the problem seem to require further exploitation for algorithmic development, which prevent them from becoming useful alternatives in practice at this moment.

1.6 Statement of The Problem

Model order reduction is one of the most widely encountered dynamical systems. A lower order controller for high order plants is desirable, for using fewer components in hardware implementation and obtaining higher reliability of controllers. The effectiveness of linear matrix inequalities ($LMI's$) in linear control system synthesis is now widely recognized. Reducing various synthesis problems can be represented by linear matrix inequalities $LMI's$, we can obtain desired controllers efficiently and reliably by solving those $LMI's$ via convex optimization algorithms [16]. In the past two decades, intensive research effort has been made to establish ways for reducing general controller synthesis problems into $LMI's$. It has been shown that such reductions are indeed possible, provided that we design state feedback or full-order controllers that satisfy a single design specification.

This work presents a modified approach for Frobenius Hankel (FH) norm model reduction method for the $MIMO$ case based on linear matrix inequality (LMI) techniques. This approach allows both continuous time and discrete time cases to be treated in a unified manner, as in [4] for H_∞ model reduction and as in [17] for H_2 model reduction. Necessary and sufficient conditions for the existence of suboptimal approximates are expressed in bilinear matrix inequality BMI form, which will then be converted to a set of $LMI's$ and a $non-convex$ matrix rank constraint. Then we will use an algorithm to solve this problem. Due to the non-convex property of the problem, the suggested method does not guarantee global convergence. Also the algorithm essentially will solves suboptimal problems and hence avoids the issue of existence/non-existence of the optimal solution. Moreover, a search will be carried out for the feasible H_2 error by executing feasible tests and therefore can be terminated when a desired difference between the achieved error and the *local* optimum is reached.

1.7 Thesis Outline

This thesis explores the solution technique used to reduce the order of a model using (LMIs) and utilizing the H_2 norm, H_∞ norm and Frobinues Hankel norm. It is organized by chapters as follows. An introduction to linear matrix inequalities(LMIs)

and the mathematical preliminaries necessary for working with the main topics of this thesis are provided in Chapter 2. Chapter 3 introduces a review of system theory and norms. The Frobinues Hankel norm is introduced in this chapter and its properties are explored. In particular, its relationship with the H_∞ and the H_2 norms are investigated. Then, we introduce the computational schemes for Cone Complementarity and Alternating Projections Algorithms for model reduction using H_∞, H_2 and H_{FH} norms in Chapter 4. Chapter 5 explores the solution and results of the proposed assumption for order model reduction. Finally, concluding remarks and comments on future research directions are presented in Chapter 6.

Chapter 2

Mathematical Preliminaries

This thesis makes use of tools and ideas from several different areas of science and mathematics. Here, in this chapter we collect the basic definitions and results so that they may be used without any detailed explanation later in the thesis. These topics are covered in depth in the listed references. Also, this chapter includes an introduction to linear matrix inequalities.

2.1 Elimination Lemma

The following lemma is about the elimination of an unknown variable K from a matrix inequality and the parameterization of one feasible K.

Lemma 2.1: *Given matrices $R \in \mathrm{S}^{m\times m}$, $U \in \mathrm{R}^{m\times l}$, $V \in \mathrm{R}^{m\times k}$. U, V have full column rank. Let $U_\perp$, $V_\perp$ denote the matrices such that [U $U_\perp$], [V $V_\perp$] are square and invertible with $U_\perp^T U = 0$, $V_\perp^T V = 0$. There exists a matrix $K \in \mathrm{R}^{l\times k}$ such that*

$$R + UKV^T + VK^TU^T < 0 \tag{2.1}$$

if and only if

$$U_\perp^T R U_\perp < 0 \;\; \textit{and} \;\; V_\perp^T R V_\perp < 0 \tag{2.2}$$

If (2.2) is satisfied and $(P_2 Q_{22}^{-1} P_2^T)^{-1}$ is well-defined, then one solution of (2.1) is given by
$K = (P_2 Q_{22}^{-1} P_2^T)^{-1}(P_1 - P_2 Q_{22}^{-1} Q_{12}^T)$,
where $[P_1 \; P_2]$=$U^T[V^{+T} \; V_\perp]$, $Q_{12} = V^+ R V_\perp$, $Q_{22} = V_\perp^T R V_\perp$. U^+, V^+ represent the pseudo-inverses of matrices U, V respectively. For the proof see [16] pages(32 -33).

2.2 Schur Complement

Schur complement is a very useful tool to modify certain linear matrix inequality constraints to a different form that is more suitable for the particular algorithm. The following theorem and additional properties of determinants could be found at [16, 18].
For all matrices $A \in \mathrm{R}^{n\times n}$, $B \in \mathrm{R}^{n\times u}$ and $D \in \mathrm{R}^{y\times u}$,

$$\begin{vmatrix} A & B \\ 0 & D \end{vmatrix} = det(A)det(D)$$

If either A or D is nonsingular, then

$$\begin{bmatrix} A & B \\ B^T & D \end{bmatrix} > 0 \Leftrightarrow \begin{cases} A > 0 \\ A - BD^{-1}B^T > 0 \end{cases}$$

$$\begin{bmatrix} A & B \\ B^T & D \end{bmatrix} > 0 \Leftrightarrow \begin{cases} D > 0 \\ D - B^TA^{-1}B > 0 \end{cases}$$

The preceding matrices are known as the *Schur Complement* of A and D, respectively.

2.3 Bounded Real Lemma

The bounded real lemma proposed by Willems, Zhou and Khargonekar [19, 20] converts the H_∞ constraint of an LTI system to an equivalent LMI condition.

Lemma 2.2:(**Bounded Real Lemma**) *Given an LTI system with its transfer function* $T(s) = D+C(sI-A)^{-1}B$. *The system is asymptotically stable and* $\|T(s)\|_\infty < \gamma$ *if and only if there exists a matrix* $X = X^T > 0$ *such that*

$$\begin{bmatrix} A^TX + XA & XB & C^T \\ B^TX & -\gamma I & D^T \\ C & D & -\gamma I \end{bmatrix} > 0 \tag{2.3}$$

For more detailed information regarding the bounded real lemma the interested reader should refer to [16, 21] also the proof of this lemma can be found in these references.

2.4 Introduction to Linear Matrix Inequalities(LMI)

A linear matrix inequality is an expression of the form

$$F(x) := F_0 + x_1F_1 + ... + x_nF_n < 0 \tag{2.4}$$

where

- $x = (x_1, ..., x_n)$ is a vector of n real numbers called the decision variables.
- $F_0, ..., F_n$ are real symmetric matrices, i.e., $F_j = F_j^T,\ for\ j = 0, ..., n.$
- the inequality < 0 in (2.4) means negative definite. That is, $u^T F(x)u < 0$ for all nonzero real vectors u. Because all eigenvalues of a real symmetric matrix are real, (2.4) is equivalent to saying that all eigenvalues $\lambda(F(x))$ are negative. Equivalently, the maximal eigenvalue $\lambda_{max}(F(x)) < 0$.

It is convenient to introduce some notation for symmetric and Hermitian matrices. A matrix A is *Hermitian* if it is square and $A = A^* = \bar{A}^\top$ where the bar denotes taking the complex conjugate of each entry in A. If A is real then this amounts to saying that $A = A^\top$ and we call A *symmetric*. The sets of all $m \times m$ Hermitian and symmetric matrices will be denoted by $\mathbb{H}^m$ and $\mathbb{S}^m$, respectively.

Definition 2.1:(Linear Matrix Inequality) *A linear matrix inequality (LMI) is an inequality*

$$F(x) < 0 \tag{2.5}$$

where F is an affine function mapping a finite dimensional vector space $\mathbb{X}$ to either $\mathbb{H}$ or $\mathbb{S}$.

2.4.1 Properties of Linear Matrix Inequalities

The linear matrix inequality (2.4) defines a *convex constraint* on x. That is, the set

$$\delta := \{x \mid F(x) < 0\}$$

of solutions of the LMI $F(x) < 0$ is convex. Indeed, if $x_1, x_2 \in \delta$ and $\alpha \in (0, 1)$ then

$$F(\alpha x_1 + (1 - \alpha)x_2) = \alpha F(x_1) + (1 - \alpha)F(x_2) \prec 0$$

where we used that F is affine and where the inequality follows from the fact that $\alpha > 0$ and $(1 - \alpha) > 0$. Although the convex constraint $F(x) < 0$ on x may seem rather special, it turns out that many convex sets can be represented in this way and that these sets have more attractive properties than general convex sets.

Definition 2.2:(System of LMIs) *A system of linear matrix inequalities is a finite set of linear matrix inequalities*

$$F_1(x) < 0,\ ...,\ F_k(x) < 0. \tag{2.6}$$

We infer that the intersection of the feasible sets of each of the inequalities (2.6) is convex. In other words, the set of all x that satisfy (2.6) is convex. The question

now arises whether or not this set can be represented as the feasibility set of another LMI. The answer is yes. Indeed, $F_1(x) < 0, ..., F_k(x) < 0$ if and only if

$$\begin{pmatrix} F_1(x) & 0 & 0 & . & & & 0 \\ 0 & F_2(x) & 0 & 0 & & & 0 \\ . & . & F_3(x) & 0 & . & . & 0 \\ & & & & & & . \\ 0 & 0 & 0 & 0 & 0 & 0 & F_k(x) \end{pmatrix} < 0$$

The last inequality indeed makes sense as $F(x)$ is symmetric (or Hermitian) for any x. Further, since the set of eigenvalues of $F(x)$ is simply the union of the eigenvalues of $F_1(x), ..., F_k(x)$, any x that satisfies $F(x) < 0$ also satisfies the system of LMIs (2.6) and vice versa. Conclude that multiple LMI constraints can always be converted to a single LMI constraint. A second important property amounts to incorporating affine constraints in linear matrix inequalities. By this, we mean that combined constraints (in the unknown x) of the form

$$\begin{cases} F(x) < 0 \\ Ax = b \end{cases}, \qquad or \begin{cases} F(x) < 0 \\ x = Bu + c, \text{ for some u} \end{cases}$$

where the affine function $F : \mathbb{R}^n \to \mathbb{S}$, matrices A and B and vectors b and c are given can be lumped in one linear matrix inequality $\hat{F}(\hat{x}) < 0$. More generally, the combined equations

$$\begin{cases} F(x) < 0 \\ x \in \mathcal{M} \end{cases} \tag{2.7}$$

where $\mathcal{M}$ is an affine set in $\mathbb{R}^n$ can be rewritten in the form of one single linear matrix inequality $\hat{F}(\hat{x}) < 0$. To do this, recall that affine sets $\mathcal{M}$ can be written as

$$\mathcal{M} = \{x \,|\, x = x_0 + m, m \in \mathcal{M}_0\}$$

with $x_0 \in \mathbb{R}^n$ and $\mathcal{M}_0$ a linear subspace of $\mathbb{R}^n$. Suppose that $\hat{n} = dim(\mathcal{M}_0)$ and let $e_1, ..., e_{\hat{n}}$. $\mathbb{R}^n$ be a basis of $\mathcal{M}_0$. Let $F(x) = F_0 + T(x)$ be decomposed. Then (2.7) can be rewritten as

$$\begin{aligned} 0 > F(x) &= F_0 + T(x_0 + \hat{n} \sum_{j=1}^{\hat{n}} x_j e_j) \\ &= F_0 + T(x_0) + \sum_{j=1}^{\hat{n}} x_j T(e_j) \\ &= \hat{F}_0 + x_1 \hat{F}_1 + ... + x_{\hat{n}} \hat{F}_n \\ &=: \hat{F}(\hat{x}) \end{aligned} \tag{2.8}$$

where $\hat{F}_0 = F_0 + T(x_0)$, $\hat{F}_j = T(e_j)$ and $\hat{x} = col(x_1, ..., x_{\hat{n}})$ are the coefficients of $x - x_0$ in the basis of $\mathcal{M}_0$. This implies that $x \in \mathbb{R}^n$ satisfies (2.7) if and only if $\hat{F}(\hat{\hat{x}}) < 0$. Note that the dimension $\hat{n}$ of $\hat{x}$ is at most equal to the dimension n of

x. A third property of LMIs is obtained from a simple exercise in algebra. It turns out to be possible to convert some non-linear inequalities to linear inequalities using Schur complement lemma which was introduced at the beginning of this chapter.

2.4.2 Applications of of Linear Matrix Inequalities

Many optimization problems in control, identification and signal processing can be formulated using linear matrix inequalities. Clearly, it only makes sense to cast these problems in an LMI setting if these inequalities can be solved in an efficient and reliable way. Since the linear matrix inequality $F(x) < 0$ defines a convex constraint on the variable x, optimization problems involving the minimization (or maximization) of a performance function $f : \delta \to \mathbb{R}$ with $\delta := \{x \,|\, F(x) < 0\}$ belong to the *class of convex optimization problems*. Casting this in the setting of the previous section, it may be apparent that the full power of convex optimization theory can be employed if the performance function F is known to be convex.
Suppose that $F : \mathbb{X} \to \mathbb{S}$ is affine. There are two generic problems related to the study of linear matrix inequalities:

1. **Feasibility**: The question whether or not there exist elements $x \in \mathbb{X}$ such that $F(x) < 0$ is called a feasibility problem. The LMI $F(x) < 0$ is called feasible if such x exists, otherwise it is said to be infeasible.

2. **Optimization:** Let an objective function $F : \delta \to \mathbb{R}$ be given and suppose that $S = \{x \,|\, F(x) < 0\}$. The problem to determine

$$V_{opt} = \inf_{x \in \delta} f(x)$$

   is called an optimization problem with an LMI constraint. This problem involves the determination of V_{opt}, the calculation of an almost optimal solution x (i.e., for arbitrary $\epsilon > 0$ the calculation of an $x \in \delta$ such that $V_{opt} \leq f(x) \leq V_{opt} + \epsilon$), or the calculation of a optimal solutions x_{opt} (elements $x_{opt} \in \delta$ such that $V_{opt} = f(x_{opt})$).

2.4.3 Solution Methods for Linear Matrix Inequalities

The problems defined in the previous subsection can be solved with efficient numerical tools. LMI can be solved by the Ellipsoid Algorithm. This algorithm is simple, numerically robust and easy to implement but may be slow for larger optimization problems. The second method for solving the LMI is the convex optimaization which is based on the Interior point methods. These methods were developed in a series of papers [22] and became of interest in the context of LMI problems in the work of Yurii Nesterov and Arkadii Nemirovskii [23].

Chapter 3

The Frobenius-Hankel Norm

3.1 Representation of Dynamic Systems

As we have shown from the control courses there are two ways to represents a linear time-invariant (LTI) system, the first involves the input-output relationship of the system in terms of first order differential equations. This is often called the **state space** representation of the system because of the state variables. The other way to represent the system through the Laplace transform of the impulse response of the system. This is called the **transfer function** of the system.

Definition 3.1: *Given the state space representation of the LTI system*

$$\begin{aligned} \dot{x} &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t). \end{aligned} \tag{3.1}$$

where $x \in \mathrm{R}^n$ is the state, $u \in \mathrm{R}^m$ is the input, $y \in \mathrm{R}^r$ is the output, the transfer function of the system is denoted by:

$$G(s) = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] \triangleq C(sI - A)^{-1}B + D. \tag{3.2}$$

Thus, through the use of expression (3.2), the transfer function of the system is represented in terms of a state space representation.

3.1.1 Series Connection

Given two systems $G_1(s)$ and $G_2(s)$

$$G_1(s) = \left[\begin{array}{c|c} A_1 & B_1 \\ \hline C_1 & D_1 \end{array}\right], \; G_2(s) = \left[\begin{array}{c|c} A_2 & B_2 \\ \hline C_2 & D_2 \end{array}\right] \tag{3.3}$$

connected together in series as shown in Figure (3.1), the resulting system can be

Figure 3.1: Series Connection.

represented as

$$G_1(s)G_2(s) = \left[\begin{array}{cc|c} A_2 & B_2C_1 & B_2D_1 \\ 0 & A_1 & B_1 \\ \hline C_2 & D_2C_1 & D_2D_1 \end{array}\right] \tag{3.4}$$

Note that this representation of the resultant system is not necessarily minimal.

3.1.2 Parallel Connection

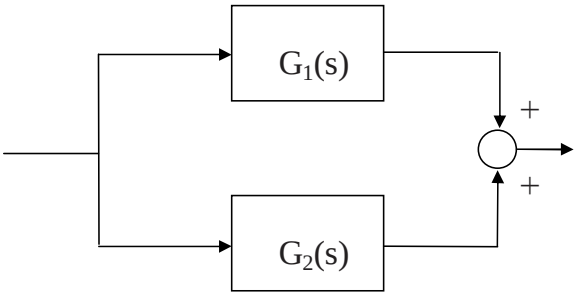

Figure 3.2: Parallel Connection.

Given two systems $G_1(s)$ and $G_2(s)$ connected in parallel as shown in Figure (3.2), the resulting system can be represented as

$$G_1(s) + G_2(s) = \left[\begin{array}{cc|c} A_1 & 0 & B_1 \\ 0 & A_2 & B_2 \\ \hline C_1 & C_2 & D_1 + D_2 \end{array}\right] \tag{3.5}$$

Note that this representation of the resultant system is not necessarily minimal.

3.1.3 Minimal Representation

The system is said to be on a minimal representation if it has no uncontrollable or unobservable states. If the system contains uncontrollable and unobservable states, it can be reduced to its minimal representation by removing the uncontrollable and

unobservable states. This can be done if the system has the following canonical form:

$$G(s) = \left[\begin{array}{cccc|c} A_{c\bar{o}} & A_{12} & A_{13} & A_{14} & B_{c\bar{o}} \\ 0 & A_{co} & A_{23} & A_{24} & B_{co} \\ 0 & 0 & A_{\bar{c}o} & A_{34} & 0 \\ 0 & 0 & 0 & A_{\bar{c}\bar{o}} & 0 \\ \hline 0 & C_{co} & C_{\bar{c}o} & 0 & D \end{array}\right] \tag{3.6}$$

This system is equivalent to the following system

$$G(s) = \left[\begin{array}{c|c} A_{co} & B_{co} \\ \hline C_{co} & D \end{array}\right] \tag{3.7}$$

Note that $A_{c\bar{o}}$ represents dynamics which are controllable but not observable, A_{co} represents dynamics which are both controllable and observable, $A_{\bar{c}o}$ represents dynamics which are observable but not controllable, $A_{\bar{c}\bar{o}}$ represents dynamics which are neither controllable nor observable.

3.2 Norms of Dynamic Systems

In this section we will introduce the most common norms used for dynamic systems which are H_2 norm and H_∞ norm.

Definition 3.2: *The H_2 norm of $G(s) \in H_2$ is defined as*

$$\|G(s)\|_2 \triangleq [Tr\{\frac{1}{2\pi}\int_{-\infty}^{\infty} G(j\omega)^T G(j\omega) d\omega\}]^{1/2} \tag{3.8}$$

The H_2 norm is called the **quadratic norm** due to its interpretation as the integral of the square of the impulse response.

Theorem 3.1. *Given $G(s) \in H_2$ and the **impulse response** $g(t) = \mathcal{L}^{-1}[G(s)]$*

$$\|G(s)\|_2^2 = Tr \int_0^{\infty} g(t)^T g(t) dt. \tag{3.9}$$

The H_2 norm can be computed from a state space representation of the system.

Proof. Can be found in [24]. □

Theorem 3.2. *Given $G(s) \in H_2$ and the controllability and observability grammians P and Q,*

$$\|G(s)\|_2^2 = Tr\{PC^TC\} = Tr\{QBB^T\} \tag{3.10}$$

Proof. Can be found in [24]. □

Similar to the H_2 the H_∞ norm is introduced now, which is often considered as a "worst-case" norm and it is used in the applications with robustness issues since it can be used to bound the performance of a system.

Definition 3.3: *The H_∞ norm of $G(s) \in H_\infty$ is defined as*

$$\|G(s)\|_\infty \triangleq \sup_\omega \bar{\sigma}[G(j\omega)] \tag{3.11}$$

The H_∞ norm is the equivalent to the maximum gain of the system as shown in the following theorem.

Theorem 3.3. *Given $G(s) \in H_\infty$*

$$\|G(s)\|_\infty = \sup_{\omega \neq 0} \frac{\|z\|_2}{\|w\|_2} \tag{3.12}$$

where $z(t)$ is the response of $G(s)$ to the input $w(t)$.

Proof. Can be found in [24]. □

This theorem is an important robustness result since it states that if a bound is known on the norm of the input, a bound can be calculated for the norm of the output.

The computation of the H_∞ norm is a difficult task since it can not be computed directly. However, we can compute an upper bound of the H_∞ norm as follows:

Theorem 3.4. *Given $G(s) \in H_\infty$*

$$G(s) = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] \tag{3.13}$$

and $\gamma > \bar{\sigma}(D)$. Then $H_\infty < \gamma$ if and only if $\mathbf{H}$ has no imaginary eigenvalues where

$$\mathbf{H} \triangleq \begin{bmatrix} A + BR^{-1}D^TC & \gamma^{-2}BR^{-1}B^T \\ -C^TS^{-1}C & -A^T - C^TDR^{-1}B^T \end{bmatrix} \tag{3.14}$$

and $R \triangleq I - \gamma^{-2}D^TD$ and $S \triangleq I - \gamma^{-2}DD^T$

Proof. Can be found in [24] or for more detailed see convex optimization books. □

For the case where the system is strictly proper, the theorem can be simplified as follows:

Theorem 3.5. *Given* $G(s) \in H_2$

$$G(s) = \left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array}\right] \tag{3.15}$$

Then $\|G(s)\|_\infty < \gamma$ *if and only if* $\mathbf{H}$ *has no imaginary eigenvalues where*

$$\mathbf{H} \triangleq \begin{bmatrix} A & \gamma^{-2}BR^{-1}B^T \\ -C^TC & -A^T \end{bmatrix} \tag{3.16}$$

Equivalently $\|G(s)\|_\infty < \gamma$ *if and only if there exists an* $X \geq 0$ *which satisfies the Riccati equation*

$$A^TX + XA + \gamma^{-2}XBB^TX + C^TC = 0 \tag{3.17}$$

Thus, the H_∞ norm of a system can be computed in iterative way be applying a search algorithm to the problem of finding the smallest γ which satisfies the bound $\|G(s)\|_\infty < \gamma$. This γ is the smallest H_∞ norm of the system.

3.3 The Frobenius-Hankel Norm

Medanic and Perkins introduced the Frobenius-Hankel (FH) norm for the design of control systems in 1987 [25]. The FH norm is defined as the Frobenius norm on the Hankel singular values. The importance of this norm comes from its relationship to more widely known norms such as H_2 and H_∞ and its good computational properties which make it suitable for use in optimization procedures.

In this section, the properties of the Frobenius-Hankel will be explored. In particular both a time-domain and frequency-domain interpretation will be made and a simple computational method will be shown for calculating the FH norm. The FH norm will also be directly related to both the H_2 and H_∞ norms.

In the following chapters the FH norm will be used as the basis for our optimization problem and applied to a model reduction problem using LMI.

Definition 3.3: *The Frobenius Hankel norm of* $G(s)\epsilon H_2$ *is*

$$\|G(s)\|_F \triangleq [\sum_{i=1}^{n} \sigma_i^2(G(s))]^{1/2} \tag{3.18}$$

3.3.1 Properties of the FH Norm

The FH norm can be easily computed directly from the grammians P and Q.

Theorem 3.6. *Given the system $G(s) \epsilon H_2$ and the controllability and observability grammians P and Q resp., then is:*

$$\|G(s)\|_F^2 = Tr\{PQ\} \tag{3.19}$$

Proof. Can be found in appendix A. □

As we see from the last expression, it involves the solution of two Lyapunov equations but avoids computation of the Hankel singular values. This expression for the FH norm is important because the trace function is more well-behaved than the singular values function.

3.3.2 Time Domain Properties of the FH Norm

The Time domain interpretation of the FH norm is as follows:

Theorem 3.7. *Given the system $G(s) \epsilon H_2$ and the impulse response of the system $g(t)$ then:*

$$\|G(s)\|_F^2 = Tr \int_0^\infty tg(t)^T g(t)dt. \tag{3.20}$$

Proof. Can be found in appendix A. □

Note the similarity of this expression to one given in Theorem 3.6. In fact, a comparison of the two expressions shows that the FH norm is in fact equivalent to a time weighted H_2 norm.

3.3.3 Frequency Domain Properties of the FH Norm

The Frequency domain interpretation of the FH norm is as follows:

Theorem 3.8. *Given the system $G(s)\epsilon H_2$ and the frequency response of the system $G(j\omega) = G(s)|_{s=j\omega}$ then:*

$$\|G(s)\|_F^2 = \frac{j}{2\pi} Tr \int_{-\infty}^{\infty} \frac{dG(j\omega)}{d\omega} G(j\omega)^* d\omega. \tag{3.21}$$

Proof. Can be found in appendix A. □

Note that this expression shows that there are two components to the FH norm. The first component comes from the $\frac{dG(j\omega)}{d\omega}$ term in the above expression which implies that the FH norm weights the "flatness" of the transfer functions. the second component is due to the $G(j\omega)$ term in the above expression which implies that the FH norm also weights the magnitude of the transfer function.

3.3.4 Relationships with Other Norms

The FH norm can be related to the H_∞ norm through the Hankel singular values of the system. In the following two theorems, expressions are shown which relate the two norms using the Hankel singular values.

This first result defines an interval in which both the H_∞ and FH norm must lie. Thus, results which apply to the H_∞ norm can be applied to the FH norm through this result.

Theorem 3.9. *Given the system $G(s)\epsilon H_2$ then*

$$\bar{\sigma}(G(s)) \leq \|G(s)\|_\infty \leq 2\sum_{i=1}^{n} \sigma_i(G(s)). \tag{3.22}$$

$$\bar{\sigma}(G(s)) \leq \|G(s)\|_{FH} \leq \sum_{i=1}^{n} \sigma_i((G(s)). \tag{3.23}$$

Proof. Can be found in appendix A. □

From the previous result, the H_∞ norm can be bounded in term of the FH norm which will give a measure of the closeness of the H_∞ norm to the FH norm.

Theorem 3.10. *Given the system $G(s)\epsilon H_2$, then*

$$\frac{1}{\sqrt{n}}\|G(s)\|_{FH} \leq \|G(s)\|_\infty \leq 2\sqrt{n}\|G(s)\|_{FH}. \tag{3.24}$$

Proof. Can be found in appendix A. □

These expressions establish the robustness properties of the FH norm through the H_∞ norm. Thus, since the computation of the FH norm is much simpler, we can use the FH norm for the analysis of the robust systems.

The Frobenius-Hankel norm can also be related to the sensitivity of the H_2 norm to a shift of the eigenvalues of the system along the real axis. Such a shift in the eigenvalues can be represented be letting A be given by:

$$A(\alpha) = A_0 + \alpha I \tag{3.25}$$

Thus, $\lambda(A) = \lambda(A_0) + \alpha$ and the sensitivity of the H_2 norm to a shift in the eigenvalues along the real axis is given by $\frac{d}{d\alpha}\|G(s)\|_2^2$. The following theorem relates this sensitivity to the FH norm.

Theorem 3.11. *Let the eigenvalues of the system $G(s)\epsilon H_2$ be given by $\lambda_i = \bar{\lambda}_i + \alpha$, then*

$$\frac{d}{d\alpha}\|G(s)\|_2^2 \mid_{\alpha=0} = 2\|G(s)\|_{FH}^2. \tag{3.26}$$

Proof. Can be found in appendix A. □

We can conclude from the last expression, the FH norm establishes a sensitivity measure for the H_2 norm with respect to a change in the relative stability of the system.

Chapter 4

Model Order Reduction Using LMI

4.1 Introduction

High-order controllers increase the hardware complexity for hardwired controllers, which directly increases cost. For digital control, high-order controllers increase the speed of the required processor and the sampling rate which will increases the cost of the processors, A/D convertors and D/A convertors. So, there has been a significant interest in the model reduction problem, namely, the problem of approximating a high order complex system by a lower order system. Model order reduction techniques aim to reduce problem complexity while simultaneously maintaining suitable solution accuracy. A large number of techniques are available for deriving reduced order models and lower order controllers. A number of approaches,e.g., [7, 9], use first order necessary conditions for optimality and develop optimization algorithms to find solutions to resulting nonlinear equations. Most of the methods in this direction are only applicable to the single input single output *(SISO)* case. Furthermore, it is argued [10, 11] that whether the global optimum is always achievable is unclear in the continuous time case (while it is shown to exist in the discrete time case) and that, in the case of non-existence of the optimum, these approaches can only find local optima which may be far from the true global optimum. Although, the existence of the global optimum is guaranteed, optimization methods based on search algorithms can have difficulties. There may be another local optima and it is difficult to guarantee that the obtained solution is the closest to the global optimum. Moreover, there is no guarantee that the chosen stopping criterion for such a search algorithm is appropriate. To overcome these problems, several algorithms based on algebraic methods have been proposed that directly solve a set of nonlinear equations [15]. These approaches seem to have potential in cases where the optima are achievable, but computation cost required for such approaches is still high, which make these methods not useful alternatives.

There are different approaches has been developed at the last decade such as [7, 9], these approaches are proposed to solve slightly modified problems for the continuous time case, where the global optimum is proven to exist and the use of a search algorithm makes sense. Those methods can deal with the multi input

multi output (*MIMO*) case and thus favorable compared to many other methods. *A problem of those methods* may be the difficulty of measuring the conservativeness of the obtained result due to the modification of the problem.

Recently, new approaches based on $LMI's$ has been developed as in [3, 4, 17]; it is emphasized that those methods deal with the original problem rather than a modified one. Also the algorithm essentially solves suboptimal problems and hence avoids the issue of existence/non-existence of the optimal solution. Those methods can deal with the *MIMO* case and thus favorable compared to many other methods in this respect. *A problem of those methods* is the use of H_2 norm which deals with the average gain and the use of H_∞ which deals with the worst case gain, in our research we will use the FH norm because of its good computational properties which make it suitable for use in optimization problems.

4.2 H_∞ Model Reduction

4.2.1 Introduction

In this section we define the H_∞ optimal model reduction problem for a stable system G of McMillan degree n with q inputs and p outputs, which can be defined as finding a stable system $\hat{G}$ of McMillan degree $\hat{n}(< n)$ with the same numbers of inputs and outputs such that the H_∞ norm of the error $\|G - \hat{G}\|_\infty$ is small. Under the same condition, the H_∞ suboptimal model reduction problem is stated as: Given $\gamma(> 0)$, find, if it exists, $\hat{G}$ that achieves the H_∞ error less than γ, i.e., achieves $\|G - \hat{G}\|_\infty < \gamma$. Assuming that both G and $\hat{G}$ are strictly proper.

4.2.2 Continuous-time H_∞ Model Reduction

Firstly let us define the state space realizations of the error system $E(s)$, let $G(s)$ and $\hat{G}(s)$ be

$$G(s) = \left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array}\right] \tag{4.1}$$

$$\hat{G}(s) = \left[\begin{array}{c|c} \hat{A} & \hat{B} \\ \hline \hat{C} & 0 \end{array}\right] \tag{4.2}$$

where $A \in \mathbb{R}^{nxn}$, $B \in \mathbb{R}^{nxq}$, $C \in \mathbb{R}^{pxn}$, $\hat{A} \in \mathbb{R}^{rxr}$, $\hat{B} \in \mathbb{R}^{rxq}$, $\hat{C} \in \mathbb{R}^{pxr}$. A state space realization of the error system is

$$E(s) = G(s) - \hat{G}(s) = \left[\begin{array}{cc|c} A & 0 & B \\ 0 & -\hat{A} & -\hat{B} \\ \hline C & -\hat{C} & 0 \end{array}\right] =: \left[\begin{array}{c|c} A_E & B_E \\ \hline C_E & 0 \end{array}\right] \tag{4.3}$$

The following theorem provide necessary and sufficient conditions for the solution of the γ-suboptimal H_∞ model reduction problem in terms of LMIs, and an explicit parametrization of all reduced-order models that correspond to a feasible solution.

Theorem 4.1. *There exists an $\hat{n}^{th}$-order system $\hat{G}$ to solve the γ-suboptimal H_∞ model reduction problem if and only if there exist matrices $X > 0$ and $Y > 0$ such that the following conditions are satisfied:*

$$AX + XA^T + BB^T < 0 \tag{4.4}$$

$$A^TY + YA + C^TC < 0 \tag{4.5}$$

$$\begin{bmatrix} X & \gamma I \\ \gamma I & Y \end{bmatrix} \geq 0 \tag{4.6}$$

and

$$rank \begin{bmatrix} X & \gamma I \\ \gamma I & Y \end{bmatrix} \leq n + \hat{n} \tag{4.7}$$

All γ-suboptimal $\hat{n}^{th}$-order models that correspond to a feasible matrix pair (X, Y) are given by

$$\begin{bmatrix} \hat{D} & \hat{C} \\ \hat{B} & \hat{A} \end{bmatrix} = \hat{G}_1 + \hat{G}_2 L \hat{G}_3, \tag{4.8}$$

where $L \in \mathrm{R}^{(p+\hat{n})\times(m+\hat{n})}$ is any matrix such that $\|L\| < 1$, and

$$\begin{aligned} \hat{G}_1 &= (M_1 - Q_{12}Q_{22}^{-1}M_2^T)(M_2Q_{22}^{-1}M_2^T)^{-1}, \\ \hat{G}_2 &= (-Q_{11} + Q_{12}Q_{22}^{-1}Q_{12}^T - G_1G_3^2G_1^T)^{\frac{1}{2}}, \\ \hat{G}_3 &= (-M_2Q_{22}^{-1}M_2^T)^{\frac{1}{2}}, \end{aligned} \tag{4.9}$$

where

$$\begin{aligned} M_1 &\triangleq \begin{bmatrix} 0 & 0 \\ 0 & R_x^2 \end{bmatrix}, \quad M_2 \triangleq \begin{bmatrix} 0 & I \\ R_xL_x^T & 0 \end{bmatrix}, \\ Q_{11} &\triangleq \begin{bmatrix} -\gamma^2 I & CL_xR_x \\ R_xL_x^TC^T & 0 \end{bmatrix}, \quad Q_{12} \triangleq \begin{bmatrix} CX & D \\ R_xL_x^TA^T & 0 \end{bmatrix}, \\ Q_{22} &\triangleq \begin{bmatrix} AX + XA^T & B \\ B^T & -I \end{bmatrix} \end{aligned} \tag{4.10}$$

Where $R_x \in \mathrm{R}^{\hat{n}\times\hat{n}}$ *is an arbitrary positive-definite matrix and* $L_x \in \mathrm{R}^{n\times(\hat{n})}$ *is an arbitrary matrix factor such that*

$$L_x L_x^T = X - \gamma^2 Y^{-1}. \tag{4.11}$$

Proof. For the proof of this theorem see [4]. □

As we can see from the previous theorem the γ-suboptimal H_∞ model reduction problem is characterized as a feasibility problem of finding a pair of positive-definite matrices (X, Y) in the intersection of the constraint sets (4.4)-(4.7). The constraints (4.4)-(4.6) are convex LMIs, but the coupling constraint set (4.7) is non-convex. Numerical algorithms based on cone complementary method and alternating projections onto the constraint sets (4.4)-(4.7) will be presented to find a solution of the non-convex feasibility problem.

The solution of the optimal H_∞ model reduction problem is obtained by solving the following non-convex minimization problem:

$$\begin{aligned} &minimize_{X,Y} \quad \gamma \\ &subject \quad to \\ &\qquad (4.4) - (4.7) \end{aligned} \tag{4.12}$$

4.2.3 Discrete-time H_∞ Model Reduction

The following theorem provide necessary and sufficient conditions for the existence of a solution to the discrete-time γ-suboptimal $\|G\|_\infty$ model reduction problem and a state-space parametrization of all reduced order models [4].

Theorem 4.2. *There exists an* $\hat{n}^{th}$*-order system* $\hat{G}$ *to solve the discrete-time* γ*-suboptimal* H_∞ *model reduction problem if and only if there exist matrices* $X > 0$ *and* $Y > 0$ *such that the following conditions are satisfied:*

$$X - AXA^T - BB^T < 0 \tag{4.13}$$

$$Y + A^T Y A + C^T C < 0 \tag{4.14}$$

$$\begin{bmatrix} X & \gamma I \\ \gamma I & Y \end{bmatrix} \geq 0 \tag{4.15}$$

and

$$rank \begin{bmatrix} X & \gamma I \\ \gamma I & Y \end{bmatrix} \leq n + \hat{n} \tag{4.16}$$

All γ-suboptimal $\hat{n}^{th}$-order models that correspond to a feasible matrix pair (X, Y) are given by

$$\begin{bmatrix} \hat{D} & \hat{C} \\ \hat{B} & \hat{A} \end{bmatrix} = \hat{G}_1 + \hat{G}_2 L \hat{G}_3, \tag{4.17}$$

where $L \in \mathrm{R}^{(p+\hat{n})\times(m+\hat{n})}$ is any matrix such that $\|L\| < 1$, and

$$\begin{aligned} \hat{G}_1 &= -(\Gamma^T \Phi \Gamma)^{-1} \Gamma^T \Phi \Theta R \Lambda^T (\Lambda R \Lambda^T)^{-1} \\ \hat{G}_2 &= (\Gamma^T \Phi \Gamma)^{-\frac{1}{2}}, \\ \hat{G}_3 &= (\Omega - \Omega \Lambda R \Theta^T [\Phi - \Phi \Gamma (\Gamma^T \Phi \Gamma)^{-1} \Gamma^T \Phi] \Theta R \Lambda^T \Omega)^{\frac{1}{2}}, \end{aligned} \tag{4.18}$$

where

$$\begin{aligned} &\Phi \triangleq (Q - \Theta R \Theta^T + \Theta R \Lambda^T (\Lambda R \Lambda^T)^{-1} \Lambda R \Theta^T)^{-1}, \\ &Q \triangleq \begin{bmatrix} \bar{X} & 0 \\ 0 & \gamma^2 I \end{bmatrix}, \quad R \triangleq \begin{bmatrix} \bar{X} & 0 \\ 0 & I \end{bmatrix}, \quad \Omega \triangleq \begin{bmatrix} I & 0 \\ 0 & X_c^{-1} \end{bmatrix}, \\ &\bar{X} \triangleq \begin{bmatrix} X & X_{pc} \\ X_{pc}^T & X_c \end{bmatrix}, \quad \Lambda \triangleq \begin{bmatrix} 0 & 0 & I \\ 0 & I & 0 \end{bmatrix}, \\ &\Theta \triangleq \begin{bmatrix} A & 0 & B \\ 0 & 0 & 0 \\ C & 0 & D \end{bmatrix}, \quad \Gamma \triangleq \begin{bmatrix} 0 & 0 \\ 0 & I \\ -I & 0 \end{bmatrix} \end{aligned} \tag{4.19}$$

and X_{pc}, X_c are arbitrary matrices such that $\bar{X} > 0$

Proof. Can be found in [4]. □

4.2.4 Computational techniques for H_∞ model reduction Using Alternating projection method

Alternating projection methods have been used in the past to solve statistical estimations and image restoration problems. They provide iterative schemes for finding a feasible point in the intersection of a family of convex sets. The basic idea is that

of a cyclic sequence of projections onto the constraint sets. Recently alternating projections have been used in control design and model and controller order reduction problems. In this chapter the alternating projection method [26] is employed for finding reduced order models. We will consider a special case of alternating projection method which can deal with a family of closed convex sets, in our case we have a pair of sets $\mathcal{C}_1$ and $\mathcal{C}_2$ in the space $S_n \times S_n$ and assume that the intersection of these sets is non-empty set. The feasibility problem of finding an element in the intersection $\mathcal{C}_1 \cap \mathcal{C}_2$ is considered. Let $P_{C_i}(X)$ denotes the orthogonal projection of X onto the set $\mathcal{C}_i$, which is equivalent to that, the matrix in C_i which has minimum distance from the matrix X. Now from the general case of alternating projection method we can define our case, suppose that $\mathcal{C}_1$ and $\mathcal{C}_2$ are closed and convex. Then, starting from any element $(X_0,\, Z_0)$ in the space, the sequence of alternating projections

$$\begin{aligned}
(X_1,\, Z_1) &= P_{C1}(X_0,\, Z_0) \\
(X_2,\, Z_2) &= P_{C2}(X_1,\, Z_1) \\
&\vdots \\
(X_{2m-1},\, Z_{2m-1}) &= P_{C1}(X_{2m-2},\, Z_{2m-2}) \\
(X_{2m},\, Z_{2m}) &= P_{C2}(X_{2m-1},\, Z_{2m-1}) \\
&\vdots
\end{aligned}$$

always converges to an element in the intersection $\mathcal{C}_1 \cap \mathcal{C}_2$ provided that the intersection is non-empty. If we have $\mathcal{C}_1$ or $\mathcal{C}_2$ is non-convex then global convergence is not guaranteed. However, if a starting point is in a neighborhood of a feasible solution local convergence is guaranteed, the alternating projection method can yield a sequence converging to an element in the intersection. In the case of the H_∞ model reduction problem, $\mathcal{C}_1$ can be taken as

$$\mathcal{C}_1 = \{(X,\, Z) | X \in S_n,\, Z \in S_n, (4.4), (4.5), (4.6)\}$$

in the continuous time case, and

$$\mathcal{C}_1 = \{(X,\, Z) | X \in S_n,\, Z \in S_n, (4.13), (4.14), (4.15)\}$$

in the discrete time case. Also,

$$\mathcal{C}_2 = \{(X,\, Z) | X \in S_n,\, Z \in S_n,\, (4.7)\, or\, (4.16)\}$$

for either case. Note that $\mathcal{C}_1$ is convex while $\mathcal{C}_2$ is not. By equipping the space $S_n \times S_n$ with the inner product

$$\langle (X_1,\, Z_1)(X_2,\, Z_2) \rangle = Tr\{X_1 X_2\} + Tr\{Z_1 Z_2\}$$

the orthogonal projection of $(X_0,\, Z_0) \in S_n \times S_n$ onto $\mathcal{C}_1$ can be found by solving the following *convex* optimization problem [26]:

$$
\begin{aligned}
&minimize \quad Tr\{S+T\} \\
&subject \quad to \quad \left[\begin{array}{cc} S & (X-X_0) \\ (X-X_0) & I \end{array} \right] \geq 0, \\
&\left[\begin{array}{cc} T & (Z-Z_0) \\ (Z-Z_0) & I \end{array} \right] \geq 0, \\
&(X,Z) \in \mathcal{C}_1, \quad S,T \in S_n.
\end{aligned}
$$

Since $\mathcal{C}_2$ is not convex, there may be more than one matrix pair that solves the projection $P_{C2}(X_0,\, Z_0)$. So, we can find more than one matrix pair that minimize the distance from $(X_0,\, Z_0)$. From [[27], Section 7.4], let $Z_0 = U\Sigma V^*$ be a singular value decomposition of Z_0. Then a projection of $(X_0,\, Z_0)$ onto $\mathcal{C}_2$ is given by

$$P_{C2}(X_0,\, Z_0) = (X_0,\, U\Sigma_r V^*)$$

where Σ_r is a diagonal matrix obtained from Σ by replacing the $(n-r)$ smallest diagonal elements of Σ by zero.
Suggested algorithm for γ-suboptimal H_∞ model reduction.

Model Reduction Scheme

1-Find $X,\, Z \in S_n$ and an H_∞-norm bound that satisfy (4.4)-(4.7) in the continuous time case (resp., (4.13)-(4.16) in the discrete time case).
2-Reduce γ Find $X,\, Z \in S_n$ that satisfy (4.4)-(4.7) (resp., (4.13)-(4.16)) using the alternating projection method, taking $(X,\, Z)$ from the previous step as a starting point.
3-If successful, go back to Step 2. Otherwise, compute an approximant from the best $(X,\, Z)$ available.

A bisection approach can be used to seek for the minimum H_∞ norm bound γ that solves the optimal H_∞ model reduction problem (4.12). If the alternating projection algorithm in Step 2 does not converge, then the value of γ should be increased. Note that the algorithm does not guarantee convergence to a global solution. When a feasible solution $(X,\, Y)$ is found, all H_∞ reduced-order models that correspond to this solution can be obtained from the parametrization (4.8)-(4.10) for the continuous-time problem or (4.17)-(4.19) for the discrete-time problem. A balanced truncation or a Hankel model reduction method can be used to obtain the initial values for $(X,\, Y)$ and γ in Step 1. For the continuous-time case the following LMI should be solved for $\bar{Y}$:

$$
\left[\begin{array}{ccc} \bar{Y}(\bar{A}+\bar{B}\bar{G}\bar{M})+(\bar{A}+\bar{B}\bar{G}\bar{M})^T\bar{Y} & \bar{Y}(\bar{D}+\bar{B}\bar{G}\bar{E}) & (\bar{C}+\bar{H}\bar{G}\bar{M})^T \\ (\bar{D}+\bar{B}\bar{G}\bar{E})^T\bar{Y} & -\gamma^2 I & \bar{F}+\bar{H}\bar{G}\bar{E} \\ (\bar{C}+\bar{H}\bar{G}\bar{M}) & \bar{F}+\bar{H}\bar{G}\bar{E} & -I \end{array} \right] < 0 \quad (4.20)
$$

where

$$
\bar{A} = \left[\begin{array}{cc} A & 0 \\ 0 & 0 \end{array} \right], \quad \bar{B} = \bar{M} = \left[\begin{array}{cc} 0 & 0 \\ 0 & I \end{array} \right], \quad \bar{E} = \left[\begin{array}{c} I \\ 0 \end{array} \right], \qquad (4.21)
$$

$$\bar{C} = \begin{bmatrix} C & 0 \end{bmatrix}, \quad \bar{H} = \begin{bmatrix} -I & 0 \end{bmatrix}, \quad \bar{F} = D, \tag{4.22}$$

and the matrix

$$\bar{G} = \begin{bmatrix} \hat{D} & \hat{C} \\ \hat{B} & \hat{A} \end{bmatrix}, \tag{4.23}$$

contains the reduced-order model from the balanced truncation or the Hankel model reduction. The initial values of X and Y are obtained from the 1-1 blocks of the matrices $\bar{X}$ and $\bar{Y}$, respectively, where

$$\bar{X} = \gamma^2 \bar{Y}^{-1}$$

The corresponding value of $\gamma = \gamma_{lb}$ or $\gamma = \gamma_{ub}$ should be used in the LMI (4.20), where γ_{lb} and γ_{ub} are will be defined in (4.25) and (4.26), respectively. For the discrete-time case, the following LMI should be solved for $\bar{X}$:

$$\begin{bmatrix} \bar{X} & 0 \\ 0 & \gamma^2 I \end{bmatrix} > \begin{bmatrix} \bar{A}+\bar{B}\bar{G}\bar{M} & \bar{D}+\bar{B}\bar{G}\bar{E} \\ \bar{C}+\bar{H}\bar{G}\bar{M} & \bar{F}+\bar{H}\bar{G}\bar{E} \end{bmatrix} \begin{bmatrix} \bar{X} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \bar{A}+\bar{B}\bar{G}\bar{M} & \bar{D}+\bar{B}\bar{G}\bar{E} \\ \bar{C}+\bar{H}\bar{G}\bar{M} & \bar{F}+\bar{H}\bar{G}\bar{E} \end{bmatrix}^T \tag{4.24}$$

where the matrices $\bar{A}$, $\bar{B}$, $\bar{C}$, $\bar{D}$, $\bar{D}$, $\bar{E}$, $\bar{M}$, $\bar{H}$, $\bar{F}$ and $\bar{G}$ are defined as before. The LMIs (4.20) and (4.24) follow from the Bounded Real Lemma [16].

4.3 H_∞ Model Reduction Algorithm Using the Cone Complementarity Method

In this section the Cone Complementarity Algorithm (CCA) which was proposed by El Ghaoui as a method to obtained reduced order controllers [28] will be implemented for solving the model order reduction problem, extensive experiments were performed using the CCA method in the solution of reduced order output feedback and static output-feedback problems. The same problem was solved using cone complementarity algorithm and methods such as: D-K iteration [30] and min-max algorithm [29], the comparison between the methods showed that in most of the cases the cone complementarity algorithm found a static controller in only one iteration and that the D-K iteration failed in the majority of the cases. This numerical study shows the effectiveness of the method in finding a solution, and also in convergence speed. Since the model reduction of dynamical systems has some similarities with the controller reduction problem and in view of the effectiveness of the CCA to handle non-convex constraints, we will customize the method to the model reduction problem. According to [28] the idea of the Cone Complementarity Algorithm

is to associate the constraint (4.7) ((4.16) for discrete system) with the minimum of $Tr(XY)$. Although, it is still a non-convex condition, it is easier to deal with compared with rank conditions. Then we have an equivalent problem formulation

$$\begin{aligned} &minimize \quad Tr(XZ) \\ &subject\,to \\ &\qquad (4.4) - (4.6) \end{aligned}$$

This problem called the *cone complementarity* problem, since it extends linear complementarity problems to the cone of positive semi definite matrices. Since the objective function is non-convex, we have to linearize the function in the neighborhood of a local point. To solve our problem using CCA, firstly find a feasible point (Xo, Yo) then a linear approximation of $Tr(XY)$ at this point will be investigated. This linear approximation of $Tr(XY)$ was proposed by Frank and Wolfe [31]. The linearization takes the form:

$$\Phi_{lin} = Tr(YoX + XoY)$$

Then we have a local searching algorithm is described as follows:

1. Find a feasible point X_0 , Y_0, if there are none, exit. Set $k = 0$.
2. Find X_{k+1}, Y_{k+1} that solve the LMI problem subject to (4.4)-(4.6)
3. If a stopping criterion is satisfied, exit. Otherwise, set $k = k + 1$ and go back to step (2)

The result of this algorithm is the solution for the suboptimal H_∞ problem, the next step is to solve the non-convex minimization problem

$$\begin{aligned} &minimize \quad \gamma \\ &subject\,to \\ &\qquad (4.4) - (4.7) \end{aligned}$$

Which gives us the solution to the optimal problem. To solve this problem a model reduction scheme based on the cone complementarity algorithm and a bisection algorithm is implemented as follows:

Model Reduction Scheme

Note that, from the hankel singular values [2] we have the upper and lower bound of the of the optimal γ_{opt} for the $\hat{n}^{th}$-order model reduction as follow:

$$\gamma_{opt} \geq \gamma_{lb} \triangleq \sigma_{k+1}(G(s)) \tag{4.25}$$

$$\gamma_{opt} \leq \gamma_{ub} \triangleq 2 \sum_{j=k+1}^{n} \sigma_j(G(s)) \tag{4.26}$$

The model reduction scheme as follows:

1. Set γ_{lb} and γ_{ub} equal to the values found in (4.25) and (4.26). Select $\hat{n} < n$, where $\hat{n}$ is the desired order of the reduced model.
2. Calculate tolerance $tol = \frac{\gamma_{ub}-\gamma_{lb}}{\gamma_{lb}}$. If $tol \leq \epsilon$ and rank $\begin{bmatrix} X & \gamma I \\ \gamma I & Y \end{bmatrix} \leq n + \hat{n}$, exit. Otherwise go to step (3).
3. Set γ equal to $\frac{\gamma_{ub}+\gamma_{lb}}{2}$, and apply Cone Complementarity Algorithm to find feasible X and Y.
4. Check X and Y resulting from the previous step, if X, Y are feasible then calculate $n_r = rank \begin{bmatrix} X & \gamma I \\ \gamma I & Y \end{bmatrix} - n$.
5. If $n_r > \hat{n}$ then set $\gamma_{ub} = \gamma$; if not, then set $\gamma_{lb} = \gamma$. Go back to step (2)

After the sub-optimal γ is found, with its associated X and Y matrices, the last step is to obtain a realization for the H_∞ reduced model using equations (4.8)-(4.10) ((4.17)-(4.19) for the discrete-time problem).

4.4 H_2 Model Reduction

In this section we define the H_2 optimal model reduction problem for a stable system G of McMillan degree n with q inputs and p outputs, which can be defined as finding a stable system $\hat{G}$ of McMillan degree $\hat{n}(< n)$ with the same numbers of inputs and outputs such that the H_2 norm of the error $\|G - \hat{G}\|_2$ is small. Under the same condition, the H_2 suboptimal model reduction problem is stated as: Given $\gamma(> 0)$, find, if it exists, $\hat{G}$ that achieves the H_2 error less than γ ($\|G-\hat{G}\|_2 < \gamma$). Assuming that both G and $\hat{G}$ are strictly proper.

4.4.1 Continuous Systems

The H_2 optimal model reduction can be defined as [11, 17]:

$$minimize\, \gamma\, (> 0)$$

$$\begin{aligned} subject \quad to \quad & A_E P + P A_E^T + B_E B_E^T < 0, && (4.27)\\ & P > 0, && (4.28)\\ & Tr\{C_E P C_E^T\} < \gamma^2 && (4.29) \end{aligned}$$

Partition P conformably with A_E and write

$$P = \begin{bmatrix} P_{11} & P_{12} \\ P_{12}^T & P_{22} \end{bmatrix}$$

where $P_{11} \in \mathcal{S}_n$, $P_{12} \in \mathbb{R}^{n\times\hat{n}}$, $P_{22} \in \mathcal{S}_{\hat{n}}$. Then, from the Schur complement formula as in section 2.2, inequality (4.27) is equivalent to

$$\begin{bmatrix} A_E P + P A_E^T & B_E \\ B_E^T & -I \end{bmatrix} =: \begin{bmatrix} AP_{11} + P_{11}A^T & AP_{12} + P_{12}\hat{A}^T & B \\ P_{12}^T A^T + \hat{A}P_{12}^T & \hat{A}P_{22} + P_{22}\hat{A}^T & \hat{B} \\ B^T & \hat{B}^T & -I \end{bmatrix} < 0 \quad (4.30)$$

Using a slack variable $W \in \mathcal{S}_p$, inequalities (4.28) and (4.29) can be expressed as:

$$Tr(W) < \gamma^2 \quad (4.31)$$

$$\begin{bmatrix} W & C_E P \\ P C_E^T & -I \end{bmatrix} =: \begin{bmatrix} W & CP_{11} - \hat{C}P_{12}^T & CP_{12} - \hat{C}P_{22} \\ P_{11}C^T - P_{12}\hat{C}^T & P_{11} & P_{12} \\ P_{12}^T C^T - P_{22}\hat{C}^T & P_{12}^T & P_{22} \end{bmatrix} > 0 \quad (4.32)$$

It is observed that neither (4.30) nor (4.32) is an LMI since there are bilinear terms such as $\hat{A}P_{12}$. In the next two theorem we will define the H_2 optimal model order reduction for stable continuous and discrete systems respectively.

Theorem 4.3. *Consider a stable continuous time system* $G(s) = \left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array}\right]$ *of McMillan degree* n*. There exists a stable continuous time system* $\hat{G}(s)$ *of McMillan degree at most* $\hat{n}$ *that satisfies* $\|G(s) - G(s)\|_2 < \gamma$ *if and only if there exist* $X, Z \in \mathcal{S}_n$ *satisfying*

$$AX + XA^T + BB^T < 0 \quad (4.33)$$

$$A(X - Z) + (X - Z)A^T < 0 \quad (4.34)$$

$$Tr\{C(X - Z)C^T\} < \gamma^2 \quad (4.35)$$

$$Z \geq 0 \quad (4.36)$$

$$rank Z \leq r \quad (4.37)$$

Proof. For the proof of theorem see [17] □

as we can see inequalities (4.33)-(4.36) are convex constraints, whereas rank constraint (4.37) is not. An optimization problem is a non-convex problem. If X and Z that satisfy (4.33)-(4.37) are found, then a reduced order model that achieves the error less than γ can be obtained by firstly using $P_{11} = X$, computing P_{12}, P_{22} from a decomposition of Z and then solving an LMI feasibility problem (4.30), (4.31), (4.32) for $\hat{A}$, $\hat{B}$, $\hat{C}$.

4.4.2 Discrete Systems

Similar to the continuous time case the discrete time case model order reduction problem of the original system $G(z)$ is the reduced order approximant $\hat{G}(z)$ that achieves the H_2 error less than γ ($\|G - \hat{G}\|_2 < \gamma$). The model reduction problem can be expressed identically to the continuous time case except for (4.27), which is to be replaced with

$$A_E P A_E^T - P + B_E B_E^T < 0. \tag{4.38}$$

Similar to the continuous time case, necessary and sufficient conditions with respect to two symmetric matrices are derived.

Theorem 4.4. *Consider a stable discrete time system* $G(z) = \left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array}\right]$ *of McMillan degree n. There exists a stable continuous time system* $\hat{G}(z)$ *of McMillan degree at most r that satisfies* $\|G(z) - G(z)\|_2 < \gamma$ *if and only if there exist* $X, Z \in \mathcal{S}_n$ *satisfying*

$$AX + XA^T - X + BB^T < 0, \tag{4.39}$$

$$A(X - Z)A^T - (X - Z) < 0 \tag{4.40}$$

$$Tr\{C(X - Z)C^T\} < \gamma^2 \tag{4.41}$$

$$Z \geq 0 \tag{4.42}$$

$$rank Z \leq r \tag{4.43}$$

For the proof of theorem see [17]. Similar to the continuous time systems, inequalities (4.39)-(4.42) are LMIs and thus convex, but the rank constraint (4.43) is not. It makes the problem a non-convex problem.

4.4.3 H_2 Model Reduction Algorithm Using the Alternating Projection Method

The following algorithm is suggested for H_2 model reduction.

1. Find X, $Z \in S_n$ and an H_2-norm bound that satisfy (4.33)-(4.37) in the continuous time case (resp., (4.39)-(4.43) in the discrete time case).

2. Reduce γ. Find X, $Z \in S_n$ that satisfy (4.33)-(4.37) (resp., (4.39)-(4.43)) using the alternating projection method, taking (X, Z) from the previous step as a starting point.

3. If successful, go back to Step 2. Otherwise, compute an approximant from the best (X, Z) available by solving a feasibility problem (4.30), (4.31) and (4.32)

We can use a bisection method with respect to γ on the implementation of our algorithm. We can conclude that, since our problem is non-convex, it is important to find a good starting point in Step 1 to have initial γ close to the optimal γ, which may be achieved by having (X, Z) close to the optimum.

4.4.4 H_2 Model Reduction Algorithm Using the Cone Complementarity Method

Similar to H_∞ problem we have an equivalent problem formulation

$$\begin{aligned} &minimize \quad Tr(XZ) \\ &subject\,to \\ &\qquad (4.33) - (4.36) \end{aligned}$$

This problem can be called the *cone complementarity* problem, to solve this problem firstly a feasible point is found (Xo, Yo) and then a linear approximation of $Tr(ZY)$ at this point will be found similar to H_∞ case:

$$\Phi_{lin} = Tr(ZoX + XoZ)$$

The local searching algorithm is described as follows:

1. Find a feasible point X_0 , Z_0, if there are none, exit. Set $k = 0$.
2. Find X_{k+1}, Z_{k+1} that solve the LMI problem subject to (4.33)-(4.36)
3. If a stopping criterion is satisfied, exit. Otherwise, set $k = k + 1$ and go back to step (2)

The result of this algorithm is the solution for the γ-suboptimal H_2 problem, the next step is to solve the non-convex minimization problem

$$\begin{aligned} &minimize \quad \gamma \\ &subject\,to \\ &\qquad (4.33) - (4.36) \end{aligned}$$

To solve this problem a model reduction scheme based on the cone complementarity algorithm and a bisection algorithm is implemented as follows:

Model Reduction Scheme

1. Set γ_{lb} and γ_{ub} equal to the values found in (4.25) and (4.26). Select $\hat{n} < n$, where $\hat{n}$ is the desired order of the reduced model.

2. Calculate tolerance $tol = \frac{\gamma_{ub}-\gamma_{lb}}{\gamma_{lb}}$. If $tol \leq \epsilon$ and rank $\begin{bmatrix} X & \gamma I \\ \gamma I & Z \end{bmatrix} \leq n + \hat{n}$, exit. Otherwise go to step (3).

3. Set γ equal to $\frac{\gamma_{ub}+\gamma_{lb}}{2}$, and apply Cone Complementarity Algorithm to find feasible X and Z.

4. Check X and Z resulting from the previous step, if X, Z are feasible then calculate $n_r = rank \begin{bmatrix} X & \gamma I \\ \gamma I & Y \end{bmatrix} - n$.

5. If $n_r > \hat{n}$ then set $\gamma_{ub} = \gamma$; if not, then set $\gamma_{lb} = \gamma$. Go back to step (2)

After the sub-optimal γ is found, with its associated X and Y matrices, the last step is to obtain a realization for the H_2 reduced model by solving a feasibility problem (4.30), (4.31) and (4.32).

4.5 H_{FH} Model Reduction

4.5.1 Introduction

In this section we define the H_{FH} optimal model reduction problem for a stable system G of McMillan degree n with q inputs and p outputs, which can be defined as finding a stable system $\hat{G}$ of McMillan degree $\hat{n}(< n)$ with the same numbers of inputs and outputs such that the H_{FH} norm of the error $\|G - \hat{G}\|_{FH}$ is small. Under the same condition, the H_{FH} suboptimal model reduction problem is stated as follows:
Given $\gamma(> 0)$, find, if it exists, $\hat{G}$ that achieves the H_{FH} error less than γ ($\|G - \hat{G}\|_{FH} < \gamma$). Assuming that both G and $\hat{G}$ are strictly proper. As we see our minimizing problem of the error $\|G - \hat{G}\|_{FH} < \gamma$ is nothing more than minimizing $(Tr\{PQ\})^{(1/2)} < \gamma$ or $Tr\{PQ\} < \gamma^2$ from Theorem 3.6 where P and Q are the grammians of the error system (4.3).

4.5.2 Continuous Systems

In this subsection the continuous time case is considered and necessary and sufficient conditions for the existence of a reduced order model achieving a specified error are derived. Then the H_{FH} optimal model reduction can be expressed as:

$$minimize \quad \gamma \quad (> 0)$$

$$\begin{aligned} subject \quad to \quad & A_E P + P A_E^T + B_E B_E^T < 0, && (4.44)\\ & A_E^T Q + Q A_E + C_E^T C_E < 0, && (4.45)\\ & P > 0, && (4.46)\\ & Q > 0, && (4.47)\\ & Tr\{PQ\} < \gamma^2 && (4.48) \end{aligned}$$

To guarantee the stability of the reduced order system $\hat{G}$ the matrices P, $Q \in \mathrm{s}_{n+\hat{n}}$ must be positive definite matrices which is equivalent to the stability of $\hat{G}$ under the assumption that (A_E, B_E) is controllable and (A_E, C_E) is observable. Partition P conformably with A_E and write

$$P = \begin{bmatrix} P_{11} & P_{12} \\ P_{12}^T & P_{22} \end{bmatrix}$$

where $P_{11} \in \mathcal{S}_n$, $P_{12} \in \mathbb{R}^{n \times r}$, $P_{22} \in \mathcal{S}_r$. Then, from the Schur complement formula as in section 2.2, inequality (4.44) is equivalent to

$$\begin{bmatrix} A_E P + P A_E^T & B_E \\ B_E^T & -I \end{bmatrix} =: \begin{bmatrix} AP_{11} + P_{11}A^T & AP_{12} + P_{12}\hat{A}^T & B \\ P_{12}^T A^T + \hat{A} P_{12}^T & \hat{A}P_{22} + P_{22}\hat{A}^T & \hat{B} \\ B^T & \hat{B}^T & -I \end{bmatrix} < 0 \qquad (4.49)$$

Partition Q conformably with A_E and write

$$Q = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{12}^T & Q_{22} \end{bmatrix}$$

where $Q_{11} \in \mathcal{S}_n$, $Q_{12} \in \mathbb{R}^{n \times r}$, $Q_{22} \in \mathcal{S}_r$. Then, from the Schur complement formula as in section 2.2, inequality (4.45) is equivalent to

$$\begin{bmatrix} A_E^T Q + Q A_E & C_E^T \\ C_E & -I \end{bmatrix} =: \begin{bmatrix} A^T Q_{11} + Q_{11} A & A^T Q_{12} + Q_{12}\hat{A} & C^T \\ \hat{A}^T Q_{12}^T + Q_{12}^T A & \hat{A}^T Q_{22} + Q_{22}\hat{A} & -\hat{C}^T \\ C & -\hat{C} & -I \end{bmatrix} < 0 \qquad (4.50)$$

Assuming that, the error system is on balanced realization then we have $P = Q = \Sigma = Q^T$. Using a slack variable $W \in \mathcal{S}_n$, inequalities (4.46), (4.47) and (4.48) can be expressed as:

$$TrW < \gamma^2 \qquad (4.51)$$

$$\begin{bmatrix} W & \Sigma \\ \Sigma & -I \end{bmatrix} > 0 \qquad (4.52)$$

It is observed that neither (4.49) nor (4.50) is an LMI since there are bilinear terms such as $\hat{A}P_{12}$. Now those conditions are expressed with respect to two decision variables symmetric matrices by eliminating $\hat{A}$, $\hat{B}$, $\hat{C}$.
As we see from the properties of the FH norm and its relationship with H_∞ norm as given in Chapter 3 on similar way to the H_∞ model order reduction we can define the H_{FH} model order reduction for a stable continuous time system $G(s)$ of McMillan degree n as follows:
Similar to the H_2 and H_∞ the necessary and sufficient condition for reducing a stable continuous time system $G(s)$ of McMillan degree n to a stable continuous time system $\hat{G}(s)$ of McMillan degree at most $\hat{n}$ that satisfies $\|G(s) - G(s)\|_{FH} < \gamma$ as follows: The reduced system of the smallest error as possible exists if and only if there exist $X, Z \in \mathcal{S}_n$ satisfying

$$AX + XA^T + BB^T < 0 \tag{4.53}$$
$$A^T Z + ZA + C^T C < 0 \tag{4.54}$$
$$X > 0 \tag{4.55}$$
$$Z > 0 \tag{4.56}$$
$$rank \begin{bmatrix} X & \gamma I \\ \gamma I & Z \end{bmatrix} \leq n + \hat{n} \tag{4.57}$$

Inequity (4.53) is derived from equation (4.49) using elimination theorem and similarly inequality (4.54) is derived from equation (4.50) for the derivation of these equation see the proof of (Theorem 1 [17]), equations (4.55) and (4.56) are necessary condition for the controllability and observability of the reduced order system similar to the H_∞ model order reduction and the last equation is necessary of obtaining the desired order $\hat{n}$ for the reduced system. As we see inequalities (4.53)-(4.56) are convex constraints but the rank constraint (4.57) is not. This optimization problem under those constraints is a non-convex problem. If X and Z that satisfy (4.53)-(4.56) are found, then a reduced order model that achieves the error less than γ can be obtained by one of the two methods, solving an LMI feasibility problem (4.49), (4.50), (4.51) and (4.52) for $\hat{A}$, $\hat{B}$, $\hat{C}$ or by obtaining the realization for the reduced model reduction using equations (4.8)-(4.10).

4.5.3 Discrete Systems

For a stable discrete time system with state space realizations of the original system $G(z)$, the reduced order approximant $\hat{G}(z)$ and the error system $E(z)$ are given as in the continuous time case. The model reduction problem can be expressed similar to the continuous time system except for (4.44) and (4.45) which is replaced with

$$A_E P A_E^T - P + B_E B_E^T < 0. \tag{4.58}$$
$$A_E^T Q A_E - Q + C_E^T C_E < 0. \tag{4.59}$$

Similar to the continuous time case, necessary and sufficient conditions with respect to two symmetric matrices are derived.
Assume we have a stable discrete time system $G(z)$ of McMillan degree n. There exists a stable discrete time system $\hat{G}(z)$ of McMillan degree at most $\hat{n}$ that satisfies $\|G(z) - G(z)\|_{FH} < \gamma$ as follows: The reduced system of the smallest error as possible exists if and only if there exist $X, Z \in \mathcal{S}_n$ satisfying

$$AXA^T - X + BB^T < 0, \tag{4.60}$$
$$A^T ZA - Z + C^T C < 0 \tag{4.61}$$
$$X > 0 \tag{4.62}$$
$$Z > 0 \tag{4.63}$$
$$rank \begin{bmatrix} X & \gamma I \\ \gamma I & Z \end{bmatrix} \leq n + r \tag{4.64}$$

As is the case with continuous time systems, inequalities (4.60)-(4.63) are LMIs and thus convex, but the rank constraint (4.64) is not. It makes our reduction problem a non-convex problem.

4.5.4 H_{FH} Model Reduction Algorithm Using the Alternating Projection Method

The following algorithm is suggested for H_{FH} model reduction using APA:

1. Find X, $Z \in S_n$ and an H_{FH}-norm bound that satisfy (4.53)-(4.57) in the continuous time case (resp., (4.60)-(4.64) in the discrete time case).
2. Reduce γ. Find X, $Z \in S_n$ that satisfy (4.53)-(4.57) (resp., (4.60)-(4.64)) using the alternating projection method, taking (X, Z) from the previous step as a starting point.
3. If successful, go back to Step 2. Otherwise, compute an approximant from the best (X, Z) available by solving a feasibility problem (4.30), (4.31) and (4.32)

A bisection approach can be used to seek for the minimum H_{FH} norm bound γ that solves the optimal H_{FH} model reduction problem. If the alternating projection algorithm in Step 2 does not converge, then the value of γ should be increased. Note that the algorithm does not guarantee convergence to a global solution. It is important to find a nice starting point in Step 1. This is because it can determine whether an approximant which is close to the global optimum will be obtained. Also, Step 2 is not in general an inexpensive task and it is desired to have initial γ close to the optimal γ, which may be achieved by having (X, Z) close to the optimum.

4.5.5 H_{FH} Model Reduction Algorithm Using the Cone Complementary Method

Similar to H_∞ and H_2 problem we have an equivalent problem formulation

$$\begin{aligned} &minimize \quad Tr(XZ) \\ &subject\,to \\ &\qquad (4.53) - (4.56) \end{aligned}$$

To solve this problem firstly a feasible point is found (Xo, Zo) and then a linear approximation of $Tr(XZ)$ at this point will be found as:

$$\Phi_{lin} = Tr(ZoX + XoZ)$$

This searching algorithm is described as follows:

1. Find a feasible point X_0 , Z_0, if there are none, exit. Set $k = 0$.
2. Find X_{k+1}, Z_{k+1} that solve the LMI problem subject to (4.53)-(4.56)
3. If a stopping criterion is satisfied, exit. Otherwise, set $k = k + 1$ and go back to step (2)

The result of this algorithm is the solution for the γ-suboptimal H_{FH} problem, the next step is to solve the optimal problem. This is done by solving the non-convex minimization problem

$$\begin{aligned} &minimize \quad \gamma \\ &subject\,to \\ &\qquad (4.53) - (4.56) \end{aligned}$$

To solve this problem a model reduction scheme based on the cone complementarity algorithm and a bisection algorithm is implemented as follows:

Model Reduction Scheme

1. Set γ_{lb} and γ_{ub} equal to the values found in (4.25) and (4.26). Select $\hat{n} < n$, where $\hat{n}$ is the desired order of the reduced model.
2. Calculate tolerance $tol = \frac{\gamma_{ub} - \gamma_{lb}}{\gamma_{lb}}$. If $tol \leq \epsilon$ and rank $\begin{bmatrix} X & \gamma I \\ \gamma I & Z \end{bmatrix} \leq n + \hat{n}$, exit. Otherwise go to step (3).
3. Set γ equal to $\frac{\gamma_{ub} + \gamma_{lb}}{2}$, and apply Cone Complementarity Algorithm to find feasible X and Z.
4. Check X and Z resulting from the previous step, if X, Z are feasible then calculate $n_r = rank \begin{bmatrix} X & \gamma I \\ \gamma I & Z \end{bmatrix} - n$.
5. If $n_r > \hat{n}$ then set $\gamma_{ub} = \gamma$; if not, then set $\gamma_{lb} = \gamma$. Go back to step (2).

After the sub-optimal γ is found, with its associated X and Z matrices, the last step is to obtain a realization for the H_{FH} reduced model by solving a feasibility problem (4.49), (4.50) (4.51) and (4.52) or by obtaining the realization for the reduced model reduction using equations (4.8)-(4.10).

4.6 Robust Model Reduction

In this section the optimal model order reduction for polytopic uncertain linear systems is investigated to show the effectiveness of the CCA.

4.6.1 Introduction

Similar to model reduction of certain systems the H_∞ optimal model reduction problem for polytopic uncertain linear systems will be the topic of this section. We want to reduce the model complexity while maintaining an accurate approximation of the original system. The ordinary techniques such as balanced truncation [1] or optimal Hankel norm model reduction [2], are widely used to reduce the order of the state-space realization of Linear Time Invariant (LTI) systems, where as little work has been done for the model reduction of uncertain systems such as [32]. The objective of this section is to duplicate the result found by [32] for model reduction for polytopic uncertain systems using (LMI) based on the model reduction procedure proposed by El Ghaoui [4].

4.6.2 Polytopic Model Reduction Problem

Consider a polytopic uncertain linear system $\mathcal{G}$

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t). \end{aligned} \tag{4.65}$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$ and $D \in \mathbb{R}^{p\times m}$. The state-space matrices of $\mathcal{G}$, $A(t)$, $B(t)$, $C(t)$, $D(t)$ are continuous functions of time since we have some entries of these matrices as an interval, and we will define the evolve of $\mathcal{G}$ as:

$$\begin{aligned} \Omega &\triangleq Co\{ \begin{pmatrix} A_i & B_i \\ C_i & D_i \end{pmatrix}, i = 1, 2, 3, ..., L \} \text{ (”Co” refer to convex hull)} \\ &= \{ \sum_{i=1}^{L} \alpha_i \begin{pmatrix} A_i & B_i \\ C_i & D_i \end{pmatrix}, \forall \alpha_i \geq 0, \sum_{i=1}^{L} \alpha_i = 1 \} \end{aligned} \tag{4.66}$$

Note that, $\mathcal{G}$ is a set of LTV systems. Introducing a set of continuous functions:

$$\mathcal{A} \triangleq \{\alpha : \mathbf{R}_+ \to \mathbf{R}^L, \text{ such that } \alpha_i(t) \geq 0, \sum_{i}^{L} \alpha_i = 1, \forall t \in \mathbf{R}_+\}, \tag{4.67}$$

and we assume one to one correspondence between allowable LTV systems from $\mathcal{G}$ and functions $\alpha \in \mathcal{A}$. Then for any LTV system $T \in \mathcal{G}$, there exists $\alpha \in \mathcal{A}$ such that

$$T(.) = \left[\begin{array}{c|c} A(.) & B(.) \\ \hline C(.) & D(.) \end{array}\right] = \sum_{i=1}^{L} \alpha_i(.) \left[\begin{array}{c|c} A_i & B_i \\ \hline C_i & D_i \end{array}\right], \tag{4.68}$$

We will also use the notation T_α to emphasize the dependence of T on a particular $\alpha \in \mathcal{A}$. The uncertain system $\mathcal{G}$ is assumed to be quadratically stable, which means thst, there exists a matrix $P \in S^{n\times n}$, $P > 0$ such that

$$A_i^T P + P A_i < 0, \quad i = 1, 2, ... L.$$

For our model reduction problem, we want to find a polytope of the form

$$\Omega^r \triangleq Co\{ \begin{pmatrix} A_i^r & B_i^r \\ C_i^r & D_i^r \end{pmatrix}, i = 1, 2, ..., L \} \tag{4.69}$$

in other words a $\hat{n}^{th}$-order ($\hat{n} < n$) polytopic uncertain system $\mathcal{G}^{\hat{n}}$

$$\begin{aligned} \dot{x}^{\hat{n}}(t) &= A^{\hat{n}} x(t) + B^{\hat{n}} u(t)) \\ y^{\hat{n}}(t) &= C^{\hat{n}} x(t) + D^{\hat{n}} u(t). \end{aligned} \tag{4.70}$$

Where as $\mathcal{G}^{\hat{n}}$ is a good approximation of the original polytopic uncertain system $\mathcal{G}$ in the induced $\mathcal{L}_2$ norm sense. So we can conclude that, the model reduction problem of polytopic uncertain system is to find the reduced order systems 2^m where m is the number of the uncertain elements in $G(s)$. So in our problem we would like to solve the following reduction problem.

Definition 4.1: *Consider a scalar $\gamma > 0$, a quadratically stable, n^{th}-order polytopic uncertain linear system $\mathcal{G}$ defined by* (4.65) (4.66)*. If there exists a polytope $\Omega^{\hat{n}}$ (associated with a $\hat{n}^{th}$-order ($\hat{n} < n$) uncertain model $\mathcal{G}^{\hat{n}}$), such that for any continuous function $\alpha \in \mathcal{A}$*

$$\|T_\alpha - T_\alpha^i\|_{i,2} < \gamma,$$

where

$$T_\alpha(.) = \sum_{i=1}^{L} \alpha_i(.) \left[\begin{array}{c|c} A_i & B_i \\ \hline C_i & D_i \end{array}\right], \qquad T_\alpha^{\hat{n}}(.) =^{ss}= \sum_{i=1}^{L} \alpha_i(.) \left[\begin{array}{c|c} A_i^{\hat{n}} & B_i^{\hat{n}} \\ \hline C_i^{\hat{n}} & D_i^{\hat{n}} \end{array}\right],$$

then we say the Polytopic Model Reduction Problem is solvable [32].
The following theorem is the main result and provides a solution to this problem , where γ is an upper bound for the worst-case approximation error between a given polytopic uncertain system and its reduced-order models.

Theorem 4.5. *Given $\gamma > 0$, the Polytopic Model Reduction Problem is solvable if there exist positive definite matrices $X, Y \in S^{n\times n}$, such that*

$$A_iX+XA_i^T+\frac{B_iB_i^T}{\gamma}<0 \tag{4.71}$$

$$A_i^TY+YA_i+\frac{C_i^TC_i}{\gamma}<0 \tag{4.72}$$

$$\begin{bmatrix} X & I \\ I & Y \end{bmatrix} \geq 0 \tag{4.73}$$

$$rank \begin{bmatrix} X & I \\ I & Y \end{bmatrix} \leq n+\hat{n} \tag{4.74}$$

If (4.71)- (4.74) are satisfied, then the vertices of one feasible polytope Ω^r are given by

$$A_i^r = (N^TL_i^{-1}N)^{-1}[I-N^TL_i^{-1}A_i^TN], \tag{4.75}$$

$$B_i^r = -(N^TL_i^{-1}N)^{-1}N^TL_i^{-1}YB_i, \tag{4.76}$$

$$C_i^r = -C_iL_i^{-1}A_i^TN-C_iL_i^{-1}N(N^TL_i^{-1}N)^{-1}[I-N^TL_i^{-1}A_i^TN], \tag{4.77}$$

$$D_i^r = D_i-C_iL_i^{-1}[I-N(N^TL_i^{-1}N)^{-1}N^TL_i^{-1}]YB_i, \tag{4.78}$$

where $L_i = A_i^TY+YA_i$ for $i=1,2,...,L$. $NN^T = Y-X^{-1}$ and N is of full column rank $m \leq \hat{n}$. Finally, the reduced-order model $\mathcal{G}_{\hat{n}}$ associated with $\Omega_{\hat{n}}$ is quadratically stable.

4.6.3 Lower and Upper Bounds for γ_{opt}

Similar to the certain system case we can define upper and lower bounds of the error between the original and the reduced system in the following theorem.

Theorem 4.6. *Given a quadratically stable, polytopic uncertain linear system $\mathcal{G}$ defined by* (4.65) - (4.66). *Let $T_i(s)$, $i=1,2,,L$ denote those LTI systems at the vertices of the polytope Ω, and $\sigma_{k+1}(T_i(s))$ the $(k+1)^{th}$ largest Hankel singular value of $T_i(s)$. The optimal γ_{opt} of the Polytopic Model Reduction Problem is bounded below by*

$$\gamma_{opt} \geq \gamma_{lb} \triangleq \max_{i=1,2,3,....,L} \sigma_{k+1}(T_i(s)) \tag{4.79}$$

Also an upper bound of γ_{opt} is obtained as follows

$$\gamma_{opt} \leq \gamma_{ub} \triangleq 2\sum_{j=k+1}^{n}\hat{\sigma}_j \tag{4.80}$$

where $\hat{\sigma_j}$ *is a generalized Hankel singular value and* $\hat{\sigma_1} > ... > \hat{\sigma_k} > \hat{\sigma_{k+1}} > ... > \hat{\sigma_n} > 0$.

To obtain the generalized Hankel singular values of a polytopic uncertain system $\mathcal{G}$, which is defined by (4.65) and (4.66), let P, Q be any solution of LMIs:

$$A_i P + P A_i^T + B_i B_i^T < 0 \quad i = 1, 2, ..., L, \tag{4.81}$$

$$A_i^T Q + Q A_i + C_i^T C_i < 0 \quad i = 1, 2, ..., L, \tag{4.82}$$

The existence of such P, Q is guaranteed by the quadratic stability assumption of the uncertain system. Then there exists a non-singular matrix Γ such that

$$\Gamma P \Gamma^T = \Gamma^{-T} Q \Gamma^{-1} = \hat{\Sigma} = \begin{bmatrix} \hat{\Sigma_1} & 0 \\ 0 & \hat{\Sigma_2} \end{bmatrix} \tag{4.83}$$

Where

$$\hat{\Sigma}_1 = diag(\hat{\sigma_1} > ... > \hat{\sigma_k}), \qquad \hat{\Sigma}_2 = diag(\hat{\sigma_{k+1}} > ... > \hat{\sigma_n})$$

where $\hat{\sigma_1} > ... > \hat{\sigma_k} > \hat{\sigma_{k+1}} > ... > \hat{\sigma_n} > 0$. As a generalization to the concept of Hankel singular values [2] for an LTI system, we will call $\hat{\sigma_j}$, $j = 1, 2, ..., n$ generalized Hankel singular values of the uncertain system $\mathcal{G}$. Since the solution of P, Q is not unique, then in order to obtained an upper bound as small as possible, an additional minimization objective is added to the set of LMIs (4.81) to (4.82), for this particular case minimizing the $trace(P + Q)$ is used as an heuristic condition. For additional details on the γ bounds see [32].

4.6.4 Robust Model Reduction Algorithm Using the Cone Complementary Method

similar to certain system model reduction problem we have an equivalent problem formulation

$$\begin{aligned} &minimize \quad Tr(XY) \\ &subject\,to \\ &\qquad (4.71) - (4.73) \end{aligned}$$

To solve this problem, firstly, a feasible point is found (Xo, Yo) and then a linear approximation of $Tr(XY)$ at this point is treated as proposed by Frank and Wolfe [31], which linearization takes the form:

$$\Phi_{lin} = Tr(YoX + XoY)$$

Then the CCA searching algorithm is described as follows:

1. Find a feasible point X_0 , Y_0, if there are none, exit. Set $k = 0$.
2. Find X_{k+1}, Y_{k+1} that solve the LMI problem subject to (4.71)- (4.73)

3. If a stopping criterion is satisfied, exit. Otherwise, set $k = k + 1$ and go back to step (2)

The result of this algorithm is the solution for the γ suboptimal H_∞ problem, the next step is to solve the optimal problem. This is done by solving the non-convex minimization problem

$$\begin{aligned} &\textit{minimize} \quad \gamma \\ &\textit{subject to} \\ &\qquad (4.71) - (4.74) \end{aligned}$$

To solve this problem a model reduction scheme based on the cone complementarity algorithm and a bisection algorithm is implemented as follows:

Model Reduction Scheme

1. Set γ_{ub} and γ_{lb} equal to the values found in (4.79) and (4.80). Select $\hat{n} < n$, where $\hat{n}$ is the desired order of the reduced model.
2. Calculate tolerance $tol = \frac{\gamma_{ub} - \gamma_{lb}}{\gamma_{lb}}$. If $tol \leq \epsilon$ and rank $\begin{bmatrix} X & \gamma I \\ \gamma I & Y \end{bmatrix} \leq n + \hat{n}$, exit. Otherwise go to step (3).
3. Set γ equal to $\frac{\gamma_{ub} + \gamma_{lb}}{2}$, and apply Cone Complementarity Algorithm to find feasible X and Y.
4. Check X and Y resulting from the previous step, if X, Y are feasible then calculate $n_r = rank \begin{bmatrix} X & \gamma I \\ \gamma I & Y \end{bmatrix} - n$.
5. If $n_r > \hat{n}$ then set $\gamma_{ub} = \gamma$; if not, then set $\gamma_{lb} = \gamma$. Go back to step (2).

After the suboptimal γ is found, with its associated X and Y matrices, the last step is to obtain a realization for the H_∞ reduced model by solving equations (4.75), (4.76), (4.77) and (4.78)

Chapter 5

Simulation Results

After introducing the LMI model order reduction techniques using the alternating projection algorithm and the cone complementarity algorithm, in this section *MATLAB LMI Toolbox* Software Package is used for testing the previously mentioned algorithms. In the following three examples, we will firstly, in Example 1 and Example 2 compare between the APA and CCA to show which algorithm gives better approximate. In Example 3, we will implement the H_{FH} model reduction using CCA. Note that, in Example 1 and Example 3 the original system is of order 12 and we will find a reduced order systems of order 4, 5, and 6 which means that, both of the two examples have three parts.

5.1 Example 1:

In this example, the real AUTM system, a 2-input, 12-state, 2-output model of an automotive gas turbine is used, which was studied in [33]. The following is a state-space representation of this model, taken from [33]:

$$G(s) = \left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array}\right]$$

where

$$A = \begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-0.202 & -1.15 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -2.36 & -13.6 & -12.8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -1.62 & -9.14 & -9.15 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -188.0 & -111.6 & -116.4 & -20.8
\end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 1.0439 & 0 & 0 & -1.794 & 0 & 0 & 1.0439 & 0 & 0 & 0 & -1.794 \\ 0 & 4.1486 & 0 & 0 & 2.6775 & 0 & 0 & 4.1486 & 0 & 0 & 0 & 2.6775 \end{bmatrix}^T$$

$$C = \begin{bmatrix} 0.264 & 0.806 & -1.42 & -15.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4.9 & 2.12 & 1.95 & 9.35 & 25.8 & 7.14 & 0 \end{bmatrix}$$

The Hankel singular values of the systems σ_i, $i = 1, 2, 3, \ldots 12$ are

$$7.1833,\ 1.4904,\ 0.92791,\ 0.58756,\ 0.46331,\ 0.23683,\ 0.16132,$$
$$0.093582,\ 0.56596 \times 10^{-3},\ 0.20608 \times 10^{-4},\ 0.1424 \times 10^{-5},\ 0.34341 \times 10^{-7}.$$

Reduced order models of McMillan degrees 4, 5, 6 are sought. According to

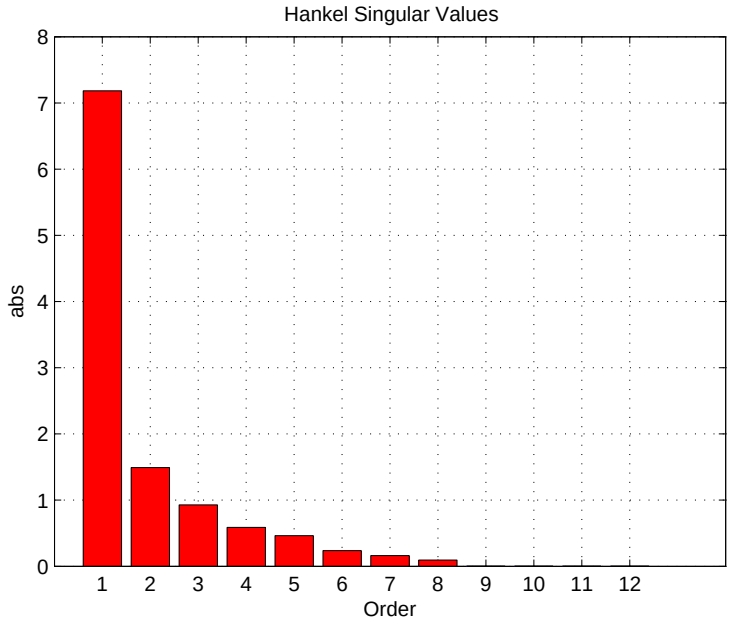

Figure 5.1: Hankel Singular Values for AUTM System For Example 1 & 3.

(4.25),(4.26):

$$\gamma_{lb} \triangleq \sigma_{k+1}(G(s)) \leq \gamma_{opt} \leq \gamma_{ub} \triangleq 2 \sum_{j=k+1}^{n} \sigma_{k+1}(G(s))$$

So for K=4 the range of the optimal γ is:

$$0.4633 \leq \gamma_{opt} \leq 1.9755 \tag{5.1}$$

For K=5 the range of the optimal γ is:

$$0.2368 \leq \gamma_{opt} \leq 1.0347 \tag{5.2}$$

For K=6 the range of the optimal γ is:

$$0.1613 \leq \gamma_{opt} \leq 0.5479 \tag{5.3}$$

5.1.1 H_∞ Model Reduction Algorithm Using CCA

For the implementation of the CCA we solve the feasibility LMI problem without the rank constraint from which we will take our starting points (Xo, Yo) then, we will start with $\gamma = \frac{\gamma_{lb}+\gamma_{ub}}{2}$ to solve our model reduction problem using CCA and we will use the bisection algorithm for next γ the following examples will illustrate the efficiency of this algorithm.
For $k = 4 \Rightarrow$ we have $\gamma = 0.5859$ and the number of iteration is 56 as shown on Figure 5.2.
For $k = 5 \Rightarrow$ we have $\gamma = 0.2509$ and the number of iteration is 34 as shown on

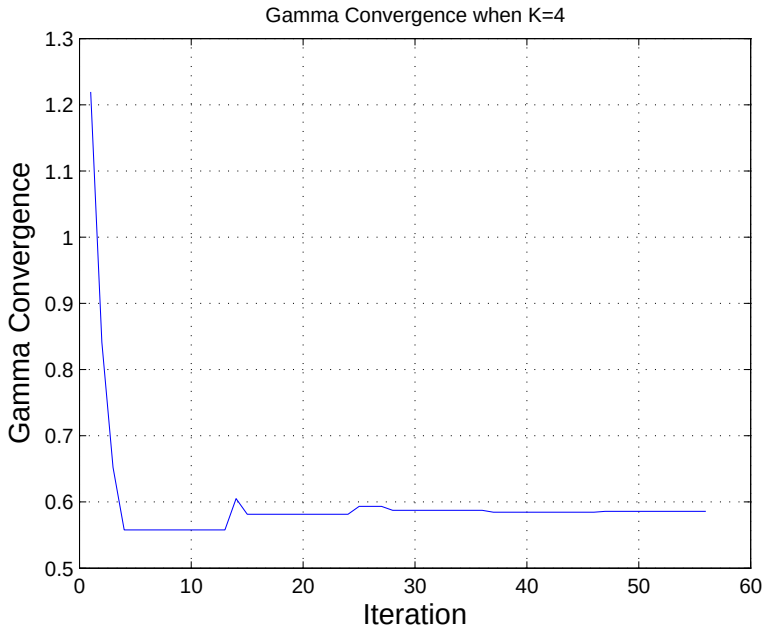

Figure 5.2: Gamma Convergence of Hinf MOR Using CCA when K=4.

Figure 5.3.
For $k = 6 \Rightarrow$ we have $\gamma = 0.1726$ and the number of iteration is 44 as shown on Figure 5.4.

5.1.2 H_∞ Model Reduction Algorithm Using APA

For the implementation of the APA, firstly, reduced-order model from the balanced truncation or the Hankel model reduction must be solved from which We will find the initial γ then We will solve (4.20) for initial (Xo,Yo). The bisection algorithm for next γ will be implemented.

For k=4
we have $\gamma = 0.9589$ and the number of iteration is 18 as shown on Figure 5.5.

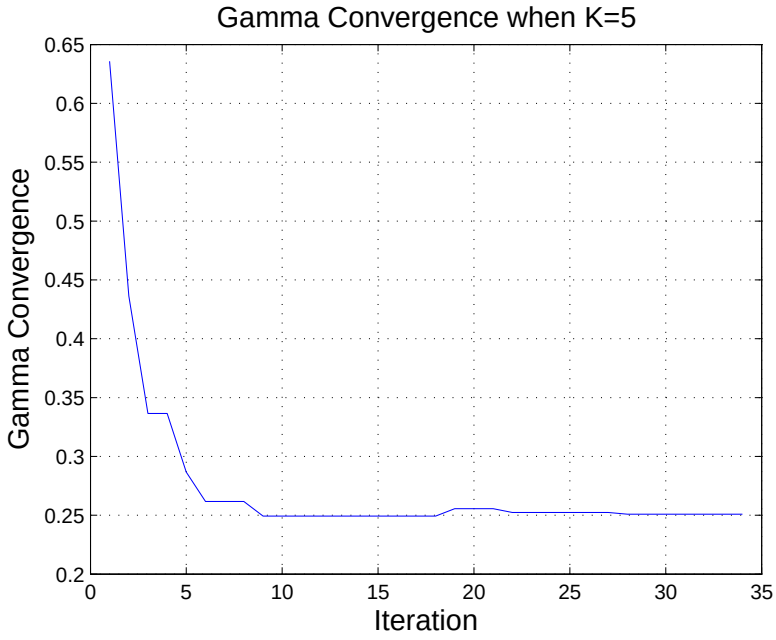

Figure 5.3: Gamma Convergence of Hinf MOR Using CCA when K=5.

For k=5
we have $\gamma = 0.4024$ and the number of iteration is 3 as shown on Figure 5.6.

For k=6 $\Rightarrow$ we have $\gamma = 0.2562$ and the number of iteration is 21 as shown on Figure 5.7.

As it is clear from the result the cone complementarity algorithm gives better results (smaller upper bound of the error) than the alternating projection method for the three cases of model order reduction. Also, as we see from the results the convergence speed of the alternating projection method is faster but we can not say that the alternating projection algorithm is faster because of the significant improvement on the upper bound of the error. On the next example we will consider an example which has been investigated using alternating projection method and we will compare it to the result we obtained using cone complementarity algorithm since in this example the system is uncertain and the convergence speed becomes important as the system uncertainties increases.

5.2 Example 2:

In this example, a polytopic uncertain plant $\mathcal{G}$, a 3-input, 4-state, 3-output model, which is studied in [34] will be investigated to demonstrate the efficiency of our H_∞

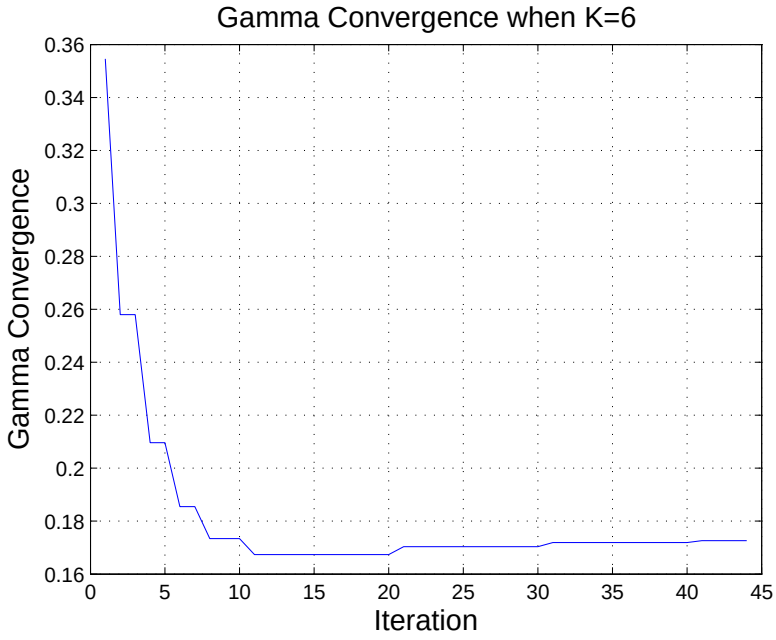

Figure 5.4: Gamma Convergence of Hinf MOR Using CCA when K=6.

model reduction algorithm using Cone Complementary Algorithm. The following is a state-space representation of this model:

$$A = \begin{bmatrix} -2 & 3 & -1 & 1 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & a_{33}(t) & 12 \\ 0 & 0 & 0 & -4 \end{bmatrix}$$

$$B = \begin{bmatrix} -2.5 & b_{12}(t) & -1.2 \\ 1.3 & -1 & 1 \\ 1.6 & 2 & 0 \\ -3.4 & 0.1 & 2 \end{bmatrix}$$

$$C = \begin{bmatrix} -2.5 & 1.3 & 1.6 & -3.4 \\ 0 & -1 & 2 & 0.1 \\ -1.2 & 1 & 0 & 2 \end{bmatrix}$$

Where $a_{33} \in [-3.5, -2.5]$ and $b_{12} \in [-0.5, 0.5]$ The Generalized Hankel singular values of the system are $\hat{\sigma}_i$, $i = 1, 2, 3, 4$ are

$$6.9034,\ 5.4239,\ 3.2328,\ 1.0424$$

We are interested on a second order approximation of the system, from theorem 4.6, the lower and upper bounds for γ_{opt} are

$$\gamma_{lb} \triangleq \max_{i=1,2,3,\ldots,L} \sigma_{k+1}(T_i(s)) = 3.14 \leq \gamma_{opt} \leq \gamma_{ub} \triangleq 2 \sum_{j=k+1}^{n} \hat{\sigma}_j = 8.47$$

We will solve the problem using the H_∞ model order reduction using cone complementarity algorithm and we will compare our results with the results obtained by

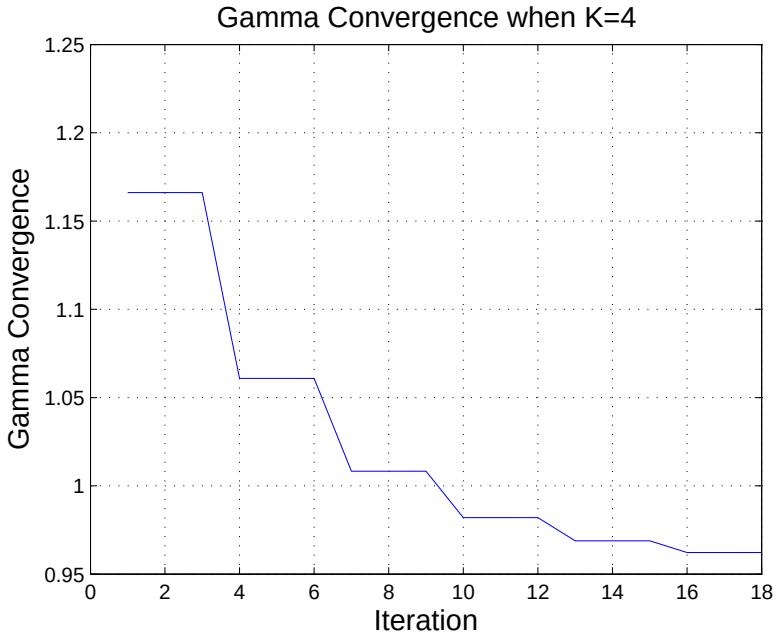

Figure 5.5: Gamma Convergence of Hinf MOR Using APA.

l_2 norm model order reduction using alternating projection method algorithm [32]. The corresponding second-order polytope obtained through the Cone complementarity method is as follows:

$$\Omega^{r,2} \triangleq Co\{ \begin{pmatrix} A_i^r & B_i^r \\ C_i^r & D_i^r \end{pmatrix},\ i = 1,\ 2,\ 3,\ 4\}$$

where

$$A_1^r = A_2^r = \begin{bmatrix} -1.8925 & 1.9635 \\ -0.2466 & -0.5270 \end{bmatrix}, \qquad A_3^r = A_4^r = \begin{bmatrix} -1.1861 & 1.2752 \\ -0.5740 & -0.2864 \end{bmatrix},$$

$$B_1^r = \begin{bmatrix} 3.8290 & -2.2179 & -3.6554 \\ -0.1263 & -0.8196 & 1.4538 \end{bmatrix}, \qquad B_2^r = \begin{bmatrix} 3.8290 & -1.6538 & -3.6554 \\ -0.1263 & -0.5373 & 1.4538 \end{bmatrix},$$

$$B_3^r = \begin{bmatrix} 3.6259 & -2.2844 & -3.4390 \\ 0.0121 & -0.7896 & 1.3544 \end{bmatrix}, \qquad B_4^r = \begin{bmatrix} 3.6259 & -1.7422 & -3.4390 \\ 0.0121 & -0.5034 & 1.3544 \end{bmatrix},$$

$$C_1^r = C_2^r = \begin{bmatrix} -1.5068 & 0.1575 \\ -1.8050 & 0.8617 \\ -0.9083 & 0.2666 \end{bmatrix}, \qquad C_3^r = C_4^r = \begin{bmatrix} -1.6017 & 0.2565 \\ -1.9214 & 0.9381 \\ -0.8263 & 0.2180 \end{bmatrix},$$

$$D_1^r = \begin{bmatrix} 3.0400 & 0.0636 & -0.3172 \\ 0.1109 & 0.1796 & -0.5675 \\ 0.3416 & -0.1381 & 0.4199 \end{bmatrix}, \qquad D_2^r = \begin{bmatrix} 3.0400 & -0.2805 & -0.3172 \\ 0.1109 & 0.2949 & -0.5675 \\ 0.3416 & -0.2816 & 0.4199 \end{bmatrix},$$

$$D_3^r = \begin{bmatrix} 3.0637 & 0.0726 & -0.3463 \\ 0.1654 & 0.1902 & -0.6027 \\ 0.3003 & -0.1455 & 0.4447 \end{bmatrix}, \qquad D_4^r = \begin{bmatrix} 3.0637 & -0.2681 & -0.3463 \\ 0.1654 & 0.3061 & -0.6027 \\ 0.3003 & -0.2890 & 0.4447 \end{bmatrix}$$

For $k = 2 \Rightarrow$ we have $\gamma = 3.3103$ and the number of iteration is 167.
The approximation errors of the second-order model at its polytope vertices are

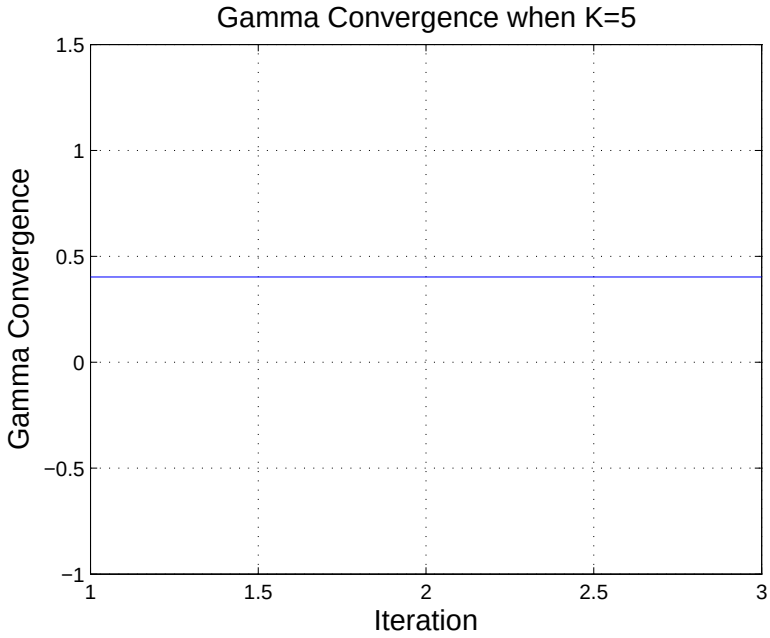

Figure 5.6: Gamma Convergence of Hinf MOR Using APA.

Table 5.1: Performance comparison between H_∞ CCA & L_2 APA for a polytopic uncertain system

Method	$n \Rightarrow K$	γ	Iterations
APA	$4 \rightarrow 2$	3.79	255
CCA	$4 \rightarrow 2$	3.3103	167

shown in Figure 5.8 and the γ convergence curve is given in Figure 5.9. Alternatively, the model reduction problem has been solved using alternating projection method [32]. The sub-optimal γ level achieved is 3.79, which is slightly worse than the cone complementarity algorithm. On the other hand, the computational scheme based on cone complementarity algorithm is much faster than alternating projection method. From the Table 5.1 the CCA method is faster than the APA approach.

5.3 Example 3:

In this example we will solve the first problem AUTM system model reduction using H_{FH} *model order reduction using CCA* according to the reduction scheme given in Chapter 4.
For k=4 $\Rightarrow$ we have $\gamma = 0.5811$ and the number of iteration is 199 as shown on Figure 5.10.

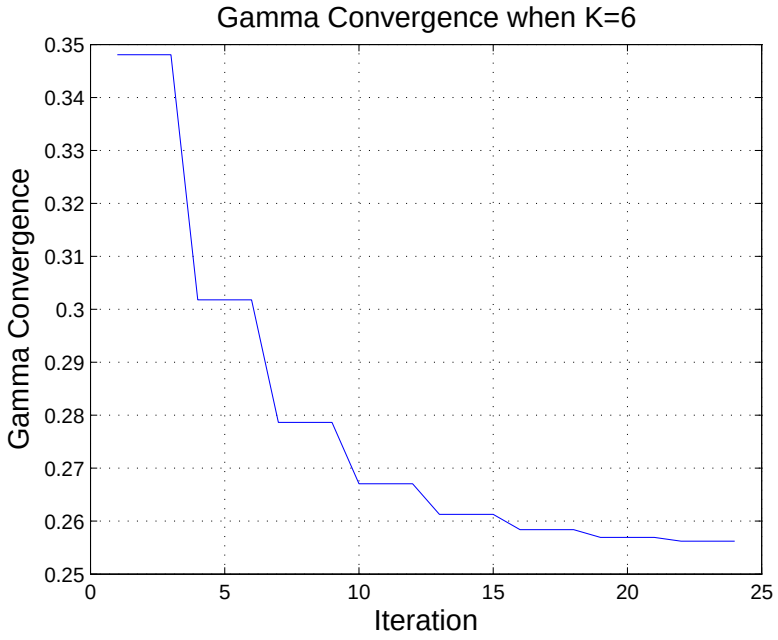

Figure 5.7: Gamma Convergence of Hinf MOR Using APA.

The reduced order system is:

$$A_r = \begin{bmatrix} -0.3820 & 0.4783 & 0.2868 & 0.4002 \\ 0.0027 & -1.6693 & 0.4012 & 1.7412 \\ -0.1845 & -0.0127 & -0.0598 & -0.5252 \\ -0.3510 & 0.7043 & 0.1647 & -1.1161 \end{bmatrix}$$

$$B_r = \begin{bmatrix} 2.8389 & 0.9943 \\ -0.3244 & 3.7565 \\ 3.9492 & 0.5817 \\ -0.2907 & -1.4660 \end{bmatrix},$$

$$C_r = \begin{bmatrix} 0.1232 & 1.1940 & -0.2259 & -1.0859 \\ 0.6279 & -0.5821 & -0.0115 & 0.0836 \end{bmatrix}$$

For $k = 5 \Rightarrow$ we have $\gamma = 0.2477$ and the number of iteration is 119 as shown on Figure 5.11.

The reduced order system is:

$$A_r = \begin{bmatrix} -0.1372 & 0.9890 & -0.2382 & 0.2155 & -0.0664 \\ -1.7142 & -1.4463 & 0.2037 & 1.0936 & -0.2285 \\ 0.6898 & 0.3873 & -1.2437 & -0.7646 & -0.6301 \\ 0.1521 & 0.0538 & -0.3034 & -0.3602 & -0.1061 \\ 0.0410 & 0.0373 & 0.2304 & 0.0358 & -0.0531 \end{bmatrix}$$

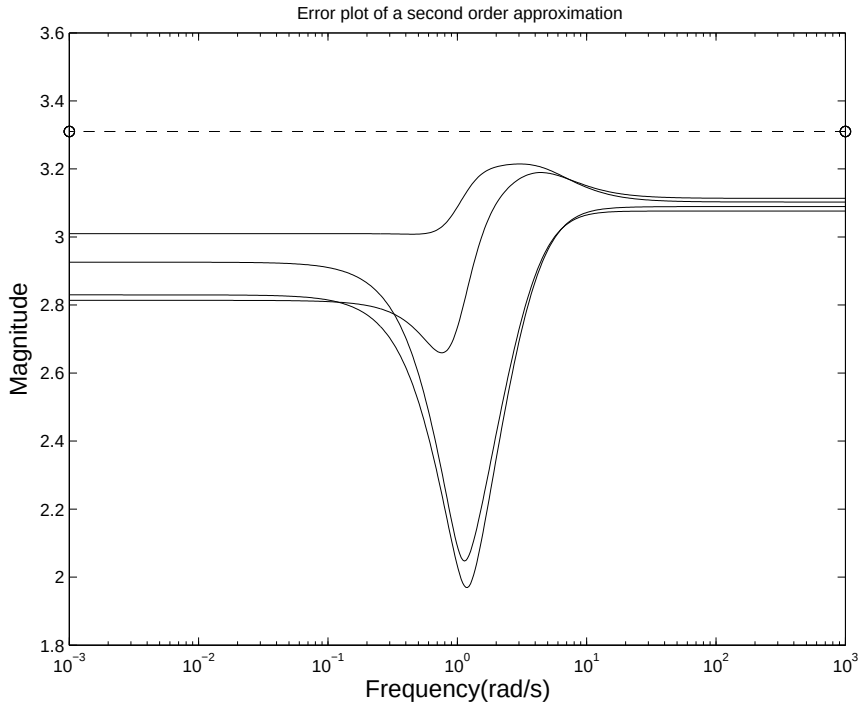

Figure 5.8: The second order approximation errors at vertices using CCA, the dashed line represents γ

$$B_r = \begin{bmatrix} 3.2138 & 1.4084 \\ -2.4190 & 1.0508 \\ -1.0266 & 3.8599 \\ 6.4252 & 2.5705 \\ -1.9309 & -0.6574 \end{bmatrix},$$

$$C_r = \begin{bmatrix} -0.0828 & -0.0509 & 0.7600 & 0.2806 & 0.1465 \\ -0.1489 & -0.8081 & 0.0920 & 0.2897 & -0.0579 \end{bmatrix}$$

For k=6 $\Rightarrow$ we have $\gamma = 0.1719$ and the number of iteration is 102 as shown on Figure 5.12.

and the reduced order system is:

$$A_r = \begin{bmatrix} -1.1031 & 1.0016 & 0.1277 & 1.0128 & -0.0389 & 3.6944 \\ -2.6459 & -0.8348 & 0.3200 & 1.5121 & -0.0593 & 5.4403 \\ 0.7966 & 0.0358 & -1.2137 & -0.7190 & -0.6867 & -2.2714 \\ 0.8158 & -0.1821 & -0.4073 & -0.7915 & -0.1075 & -3.6803 \\ 0.1101 & 0.0097 & 0.2143 & -0.0297 & -0.0517 & -0.2794 \\ 3.8645 & -0.3498 & -1.1746 & -2.6351 & -0.0996 & -13.5736 \end{bmatrix},$$

$$B_r = \begin{bmatrix} 2.3694 & -4.1339 & -0.5646 & 8.4467 & -2.3688 & 3.2309 \\ 1.2201 & 0.3114 & 5.0491 & 3.1056 & -0.7754 & 1.6663 \end{bmatrix}^T,$$

$$C_r = \begin{bmatrix} -0.1326 & -0.0060 & 0.6320 & 0.2130 & 0.1309 & 0.5531 \\ -0.5555 & -0.5311 & 0.1115 & 0.4466 & -0.0151 & 1.7789 \end{bmatrix}$$

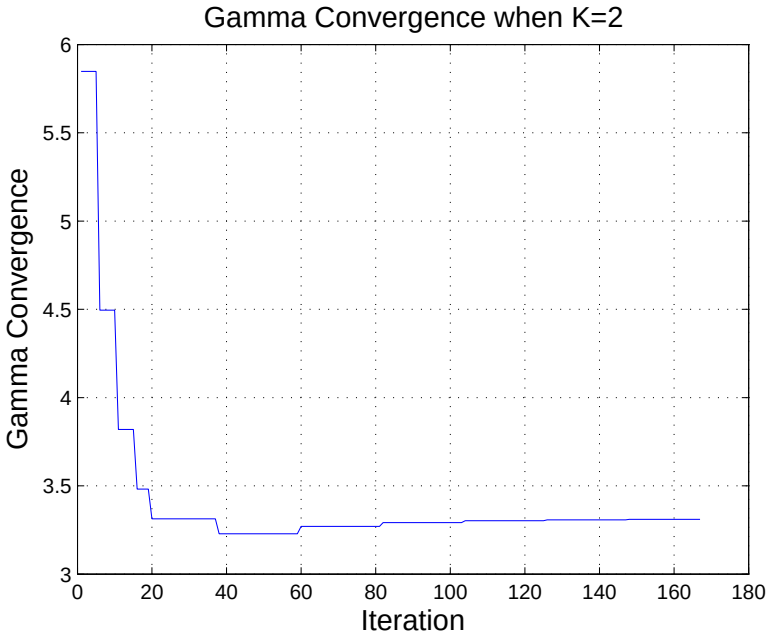

Figure 5.9: Gamma Convergence of Hinf MOR Using CCA.

Table 5.2: Performance Comparison Between H_∞ and H_{FH} for AUTM system

Norm	Method	$n \Rightarrow K$	σ_{k+1}	γ
H_∞	CCA	$12 \rightarrow 4$	0.46331	0.5859
H_{FH}	CCA	$12 \rightarrow 4$	0.46331	0.5811
H_∞	CCA	$12 \rightarrow 5$	0.23683	0.2509
H_{FH}	CCA	$12 \rightarrow 5$	0.23683	0.2477
H_∞	CCA	$12 \rightarrow 6$	0.16132	0.1726
H_{FH}	CCA	$12 \rightarrow 6$	0.16132	0.1719

Table 5.2 summarizes the comparison between the H_∞ and H_{FH} model order reduction using cone complementarity algorithm for the AUTM system for $K = 4, 5, 6$. We can conclude from this table that H_{FH} is slightly better than H_∞ but it needs more iterations to achieve the suboptimal solution. Also by the comparing of the result with the results obtained in [17] for H_2 model reduction we find an improvement on the error between the original and the reduced systems by using H_∞ and H_{FH} cone complementarity algorithm.

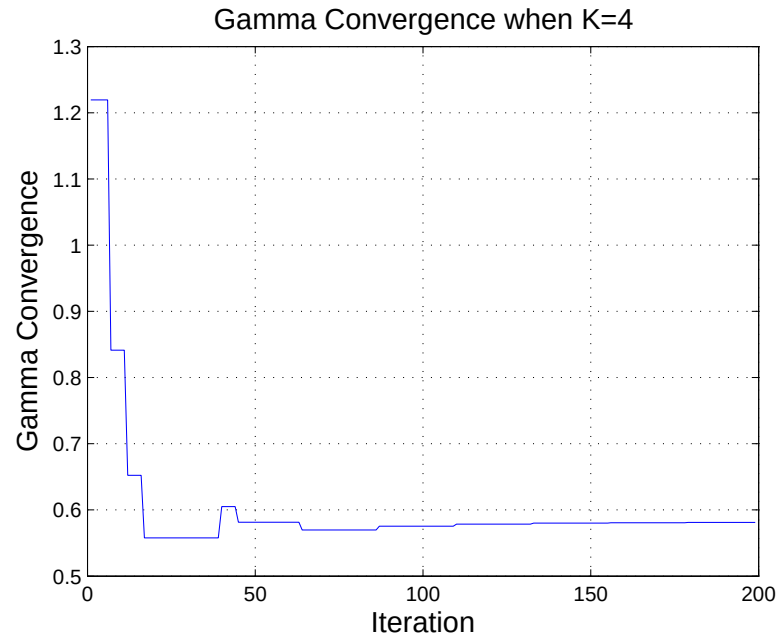

Figure 5.10: Gamma Convergence of HFH MOR Using CCA.

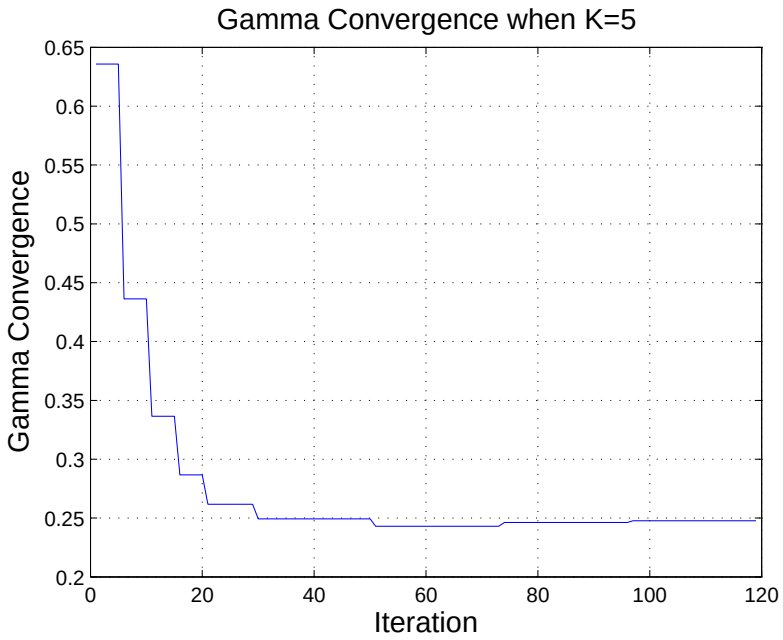

Figure 5.11: Gamma Convergence of HFH MOR Using CCA.

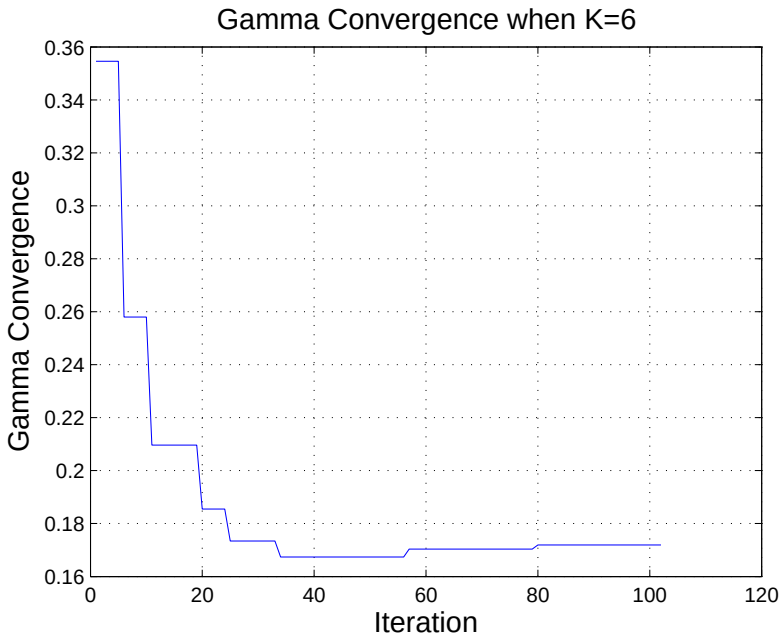

Figure 5.12: Gamma Convergence of HFH MOR Using CCA.

Chapter 6

Coclusion and Future Work

This thesis addressed the problem of optimal model order reduction which is one of the most widely problems encountered dynamical systems. There is a trade off between the system simplicity and accuracy, simple systems are inaccurate where as complex system is expensive in the implementation. A reliable lower order controller for high order plants is desirable. The effectiveness of linear matrix inequalities ($LMI's$) in linear control system synthesis is now widely recognized. Reducing various synthesis problems can be represented by linear matrix inequalities $LMI's$, we can obtain desired controllers efficiently and reliably by solving those $LMI's$ via convex optimization algorithms. We considered the H_∞, H_{FH}, H_2 model reduction problem using LMI. Necessary and sufficient conditions for the existence of a suboptimal reduced order model was derived for both continuous and discrete time cases by means of LMI techniques. The resulting constraints are quasi-convex and do not allow globally convergent algorithms to be developed. A numerical algorithms were proposed which utilized the CCA and APA. A comparison between the CCA and the APA had been investigated which proved that, the CCA gave better results but it needs more iterations for certain LTI systems. The proposed H_∞ and H_{FH} algorithm using CCA can find suboptimal approximates which are at least as good as those computed by previously proposed methods, which is demonstrated by numerical examples.

An extension for the CCA, model order reduction of polytopic uncertain systems had been outlined and implemented for H_∞ which gave results as good as those computed by other methods, again here, we compared the CCA with the APA which proved that the CCA not only can find a lower upper bound than the APA, but also its computational time is very low compared with the second method for polytopic uncertain systems. This fact made the CCA a good alternative to handle large dimensional uncertain polytopic systems. CCA has several advantages and covers both continuous and discrete time systems similar to APA. The difference in the programs is trivial.

As a future work the proposed algorithm can be extended to cover other types of uncertain system such as frequency weighted uncertain systems, also more investigation about the implantation of uncertain model reduction using H_2 norm or H_{FH} norm using CCA can be outlined. New areas of research are open in the application of the CCA to problems with quasi-convex constraints.

Bibliography

[1] S. Gugercin & A. C. Antoulas, *A survey of model reduction by balanced truncation and some new results*, International Journal of Control, 77(8) Page(s):748-766, (May 2004).

[2] K. Glover, *All optimal Hankel-norm approximations of linear multivariable systems and their L_∞-error bounds*, International Journal of Control, 39(6) Page(s):1115-1193, (June 1984).

[3] Y. Ebihara & T. Hagiwara, *On H_∞ model reduction using LMIs*, IEEE Transactions on Automatic Control, 49(7) Page(s):1187-1191,(July 2004).

[4] K. M. Grigoriadis, *Optimal H_∞ model reduction via linear matrix inequalities: Continuous and discrete-time cases*, Systems and Control Letters, 26(5) Page(s):321-333, (December 1995).

[5] V.M. Adamjan, D.Z. Arov, and M.G. Krein. *Analytic Properties of the Schmidt Pairs of a Hankel operator and the generalized Schur-Takagi problem*, Math. Sb. (N.S.), 86 Page(s):34-75, (1971).

[6] L. Baratchart, M. Cardelli & M. Olivi, *Identification and rational L_2 approximation: A gradient algorithm*, MA Journal of Mathematical Control and Information, Automatica, 27(2) Page(s):413417, (March 1991).

[7] A. Ferrante, W. Krajewski, A. Lepschy, & U. Viaro, *Convergentalgorithm for L_2 model reduction*, Automatica, 35(1) Page(s):75-79,(January 1999).

[8] L. Meier III & D. G. Luenberger, *Approximation of linear constant systems*, IEEE Transactions on Automatic Control, 12(5) Page(s):585-588, (October 1967).

[9] J. T. Spanos, M. H. Milman, & D. L. Mingori, *A new algorithm for L_2 optimal model reduction*, Automatica, 28(5) Page(s):897-909,(September 1992).

[10] X.-X. Huang, W.-Y. Yan, & K. L. Teo, *On H_2 near-optimal model reduction*, IEEE Transactions on Automatic Control, 46(8) Page(s):1279-1284,(August 2001).

[11] W. Yan & J. Lam, *An approximate approach to H_2 optimal model reduction*, IEEE Transactions on Automatic Control, 44(7) Page(s):1341-1358,(July 1999).

[12] L. Baratchart, *Existence and generic properties of L_2 approximants for linear systems*, MA Journal of Mathematical Control and Information, 3 Page(s):89-101,(1986).

[13] B. Hanzon & J. M. Maciejowski, *Constructive algebra methods for the L_2-problem for stable linear systems*, Automatica, 32(12) Page(s):1645-1657, (December 1996).

[14] D. Jibetean, *Algebraic Optimization with Applications to System Theory*, PhD thesis, Vrije University, Amsterdam, (2003).

[15] R. L. M. Peeters, B. Hanzon, & D. Jibetean, *Optimal H_2 model reduction in state-space: A case study*, Proceedings of the European Control Conference ECC 2003, Cambridge, U.K., (September 2003).

[16] S. Boyd, L. El Ghaoui, E. Feron, & V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, volume 15 of SIAM Studies in Applied Mathematics. Society for Industrial and Applied Mathematics, Philadelphia, PA, (1994).

[17] M. Kanno, *H_2 Model Reduction Using $LMIs$*, 44th IEEE Conference 12-15 Dec, Page(s):5887-5892,(2005).

[18] C. Meyer, *Matrix analysis and applied linear algebra*, Mathematics department of NCSU, Raleigh, NC, (1991).

[19] P. Khargonekar &K. Zhou. *An algegraic riccati equation approach to H_∞ optimization*, System and Control Letters 11 page(s):85-91,(1988).

[20] J.C. Willems. *Least square stationary optimal control and the algebraic riccati equation*, IEEE transaction automatic control, AC-16, Page(s):621-627, (1971).

[21] J. Doyle & K. Zhou. *Essentials of Robust Control*, Prentice-Hall, New Jersey, (1998).

[22] N. Kamarkar, *A new polynomial-time algorithm for linear programming*, Combinatorica, 4 Page(s):373-395, (1984).

[23] Yurii Nesterov and Arkadii Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*, SIAM, (1994).

[24] R. A. Ramaker. *The Design of Low Order Controllers Using the Frobenius-Hankel Norm*, PhD thesis, University of Illions at Urbana-Champaign,(1990).

[25] J. Medanic & W. R. Perkins. *Frobenius-Hankel norm framework for disturbance rejection and low order decentralized controller design* , Proceedings Fifth AFOSR Fourm on Space Structure, Page(s):77-79, (1987).

[26] K. M. Grigoriadis & E. B. Beran, *Alternating projection algorithms for linear matrix inequalities problems with rank constraints. In L. El Ghaoui and S.-I. Niculescu, editors, Advances in Linear Matrix Inequality Methods in Control*, chapter 13, page(s): 251-267, Society for Industrial and Applied Mathematics, (2000).

[27] R. A. Horn & C. R. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, (1985).

[28] L. El Ghaoui & M. Aitrami, *A cone complementary algorithm for static output-feedback and related problems*, IEEE Transaction Automatic Control, 42 Page(s):1171-1172, (1993).

[29] C. C. Souza, J. C. Geromel & R. E. Skelton, *Lmi numerical solution for output feedback stabilization*, In Proc. IEEE Conf. Decision Contr., Page(s): 40-44, (1995).

[30] T. Iwasaki& M.A. Rotea, *An alternative d-k iteration*, in Proc. Amer. Control Conf., Page(s) 53-57, (1994).

[31] O.L. Mangasarian & J.S. Sepang, *The extended linear complementarity problem*, SIAM J. Matrix Analysis Appl., 2 Page(s):359 -398, (1995).

[32] Fen Wu, *Induced L2 norm reduction of polytopic uncertain linear systems*, Automatica, 32 Page(s):14 -19, (1996).

[33] Y. S. Hung and A. G. J. MacFarlane, *Multivariable Feedback: A Quasi-Classical Approach*, volume 40 of Lecture Notes in Control and Information Sciences, Springer-Verlag, Berlin, (1982).

[34] D. Kavranoglu, *Zeroth order H_∞ norm approximation of multivariable systems*, Numer. Funct. Anal. Optim. 14 Page(s):89-101, (1993).

Appendix A

Proof of Theorems

On appendix A we provide the proofs of the theorems and definition for the completeness of this report.

Proof of Theorem 3.6:

Proof. From definition (3.2),

$$\|G(s)\|_F^2 = Tr\{\Sigma^2\} \tag{A.1}$$

where $\Sigma = diag(\sigma_1 ... \sigma_n)$. Since there exists T nonsingular such that $T^{-T}PT^{-1} = \Sigma$ and $TQT^T = \Sigma$

$$\begin{aligned} \|G(s)\|_F^2 &= Tr(T^{-T}T^T)\Sigma(TT^{-1})\Sigma \\ &= Tr(T^T\Sigma T)(T^{-1}\Sigma T^{-1}) \\ &= Tr\{PQ\}. \end{aligned} \tag{A.2}$$

□

Proof of Theorem 3.7:

Proof. From Theorem3.6

$$\|G(s)\|_F^2 = Tr\{PQ\} \tag{A.3}$$

From equations (1.9) and (1.10)

$$Tr\{PQ\} = \lim_{T\to\infty} Tr[\int_0^T e^{At}BB^T e^{A^T t}dt][\int_0^T e^{A^T\tau}C^T Ce^{A\tau}d\tau] \tag{A.4}$$

which is equivalent to

$$Tr\{PQ\} = \lim_{T\to\infty} Tr \int_0^T \int_0^T [Ce^{At(t+\tau)}B][Ce^{At(t+\tau)}B]^T dtd\tau] \tag{A.5}$$

Let $g(\tau)\tau \triangleq Ce^{A\tau}B$,

$$Tr\{PQ\} = \lim_{T\to\infty} Tr \int_0^T \int_0^T g(t+\tau)^T g(t+\tau) dt d\tau \tag{A.6}$$

$$Tr\{PQ\} = \lim_{T\to\infty} Tr \int_0^T \int_\tau^{T+\tau} g(t)^T g(t) dt d\tau \tag{A.7}$$

Let $H(\tau)\tau \triangleq \int_\tau^{T+\tau} g(t)^T g(t) dt$.

$$Tr\{PQ\} = \lim_{T\to\infty} Tr \int_0^T H(\tau) d\tau \tag{A.8}$$

Integrating by parts,

$$Tr\{PQ\} = \lim_{T\to\infty} Tr[H(\tau)\tau|_0^T \int_0^T \tau dH(\tau)] \tag{A.9}$$

$$Tr\{PQ\} = \lim_{T\to\infty} Tr[\int_0^T (T-t)g(t+T)^T g(t+T) + tg(t)^T g(t) dt] \tag{A.10}$$

In the limit as $T \to \infty$, $g(t+T) \to 0$, thus

$$Tr\{PQ\} = Tr \int_0^\infty tg(t)^T g(t) dt. \tag{A.11}$$

□

Proof of Theorem 3.8:

Proof. Applying Parseval's Theorem to (3.20) yields

$$\begin{aligned} \|G(s)\|_F^2 &= \frac{1}{2\pi} Tr \int_{-\infty}^{\infty} \mathcal{F}[tg(t)]\mathcal{F}[tg(t)]^* d\omega \\ &= \frac{1}{2\pi} Tr \int_{-\infty}^{\infty} j(\frac{dG(j\omega)}{d\omega}) G(j\omega)^* d\omega \end{aligned} \tag{A.12}$$

□

Proof of Theorem 3.9:

Proof. The first expression has been shown befor. To show the second, begin first with the upper bound.

$$\begin{aligned}(\sum_{i=1}^{n}\sigma_i)^2 &= (\sum_{i=1}^{n}\sigma_i)(\sum_{j=1}^{n}\sigma_j)\\ &= \sum_{i=1}^{n}\sum_{j=1}^{n}\sigma_i\sigma_j\\ &= \sum_{k=1}^{n}\sigma_k^2 + \sum_{i=1,i\neq j}^{n}\sum_{j=1}^{n}\sigma_i\sigma_j\\ &\geq \sum_{k=1}^{n}\sigma_k^2 = \|G(s)\|_{FH}^2\end{aligned} \tag{A.13}$$

The lower bound follows directly from

$$\|G(s)\|_{FH}^2 = \sum_{i=1}^{n}\sigma_i^2 \geq \bar{\sigma}^2. \tag{A.14}$$

□

Proof of Theorem 3.10:

Proof. To show the lower bound, first note that

$$\sum_{i=1}^{n}\sigma_i^2 \leq n\bar{\sigma}^2. \tag{A.15}$$

then it follows from (3.22) that

$$\|G(s)\|_{\infty} \geq \bar{\sigma} \geq \frac{1}{\sqrt{n}}\|G(s)\|_{FH} \tag{A.16}$$

which proofs the lower bound. To show the upper bound, note that

$$\sum_{i=1}^{n}\sigma_i = E^T\Sigma \tag{A.17}$$

where $E^T \triangleq [1\,1\,...\,1]$ and$\Sigma \triangleq [\sigma_1\,\sigma_2\,....\,\sigma_n]$. The Frobenius norm of a matrix is defined as

$$\|G(s)\|_{Fr} \triangleq [TrA^TA]^{\frac{1}{2}}. \tag{A.18}$$

Since the Frobenius norm is valid matrix norm

$$\begin{aligned}\|E^T\Sigma\|_{Fr} &= \|E^T\|_{Fr}\|\Sigma\|_{Fr} \\ &= [TrE^TE]^{\frac{1}{2}}[Tr\Sigma^T\Sigma]^{\frac{1}{2}} \\ &= \sqrt{n}\|G(s)\|_{FH}\end{aligned} \tag{A.19}$$

However, since $E^T\Sigma$ is a positive valued scalar,

$$\|E^T\Sigma\|_{Fr} = E^T\Sigma \tag{A.20}$$

Thus,

$$\sum_{i=1}^{n}\sigma_i \leq \sqrt{n}\|G(s)\|_{FH}. \tag{A.21}$$

□

Proof of Theorem 3.11:

Proof. Let

$$J \triangleq \|G(s)\|_2^2 = Tr\{PC^TC\} \tag{A.22}$$

Where

$$AP + PA^T + BB^T = 0 \tag{A.23}$$

Then

$$\frac{dJ}{d\alpha} = Tr\{P_\alpha C^TC\} \tag{A.24}$$

where $P_\alpha \triangleq \frac{dP}{d\alpha}$ satisfies

$$AP_\alpha + P_\alpha A^T + 2P = 0 \tag{A.25}$$

Let Q satisfy

$$QA + A^TQ + C^TC = 0 \tag{A.26}$$

Then, using the properties of the trace, it cab be shown that

$$Tr\{P_\alpha C^TC\} = 2Tr\{PQ\}. \tag{A.27}$$

Thus

$$\frac{dJ}{d\alpha} = 2Tr\{PQ\} = 2\|G(s)\|_{FH}^2. \tag{A.28}$$

□

Appendix B

MATLAB Codes

H_∞ Model Reduction Using APA:

Main file

```
clear all;
Erof=1e-3;
clc;
time=clock;
system;
n= size(A1,1); % number of states
G=pck(A1,B1,C1,D1);
%GG=ss(A1,B1,C1,D1);
% Desired order of the reduced system
k=6;
% Upper and lower boundaries for gamma optimal
 [du,hk]=sysbal(G);
% Calculating the lower bound
 gl=sum(hk(k+1:length(hk))); % Gamma lower bound
% Calculating the upper bound
 gu=2*sum(hk(k+1:length(hk))); % Gamma upper bound
 flag=0;
flag2=0;
itr=0;
itr_d=[];
[Xo,Yo,gama]=init1(G,k,gu);
while flag==0
  tol=(gu-gl)/gl;
     if (tol <= 0.01) & flag2==1
     gama
     flag = 1;
     else
       %gama=(gu+gl)/2;
         [lmis]=deflmi(G,gama); % See file setlmi.m
        for i=1:10
           [Xin, Yin]= truncation(Xo,Yo,G,gama,k)
           itr=itr+1;
           itr_di=[itr  gama];
           itr_d=[itr_d ;itr_di];
           Km=[Xin eye(n);
           eye(n) Yin];
            nr=rank(Km,Erof)-n
          if nr <= k
           Xop=Xin;
           Yop=Yin;
           flag1=1;
           break
          else
                 Xo=Xin;
                 Yo=Yin;
```

```
                flag1=0;
%
              end
            end
          end
      if flag1==1
        gu=gama
        flag2=1
      else
        gl=gama
        flag2=0
      end
     gama=(gu+gl)/2;
end

l= itr_d(:,1);
mm = itr_d(:,2);
figure(3);
plot(l,mm,'*');
%legend( 'Original model', 'Reduced order model','n=',num2str(n)',
'k=',num2str(k),3);
xlabel('Gamma Convergence','FontSize',14);
ylabel('Iteration','FontSize',14);
%legend( 'Original model', ' Reduced order model',4);
%title(' Gamma Convergence ',['No of iteration=',
num2str(size(l,1))] ;%,['k=',num2str(k)] );
% Function modrec reconstruct a realization for
% the reduced order system
[Gr]= modrec(G,Xop,Yop,k,gama); % See file modrec.m
% Checking process (optional)
time=etime(clock,time);
time
gama
```

System definition file:

```
A1 = [0 1 0 0 0 0 0 0 0 0 0 0 ;
    -0.202 -1.15 0 0 0 0 0 0 0 0 0 0 ;
    0 0 0 1 0 0 0 0 0 0 0 0 ;
    0 0 0 0 1 0 0 0 0 0 0 0 ;
     0 0 -2.36 -13.6 -12.8 0 0 0 0 0 0 0 ;
    0 0 0 0 0 0 1 0 0 0 0 0 ;
    0 0 0 0 0 0 0 1 0 0 0 0 ;
    0 0 0 0 0 -1.62 -9.4 -9.15 0 0 0 0 ;
    0 0 0 0 0 0 0 0 0 1 0 0 ;
    0 0 0 0 0 0 0 0 0 0 1 0 ;
    0 0 0 0 0 0 0 0 0 0 0 1;
    0 0 0 0 0 0 0 0 -188 -111.6 -116.4 -20.8];


B1= [ 0  0; 4.1486 1.0439; 0 0; 0 0; 2.6775  -1.794; 0  0; 0  0;
4.1486 1.0439 ;0  0; 0  0; 0  0; 2.6775 -1.794];
C1= [0.2640 0.8060 -1.420 -15.0 0 0 0 0 0 0 0 0;
    0 0 0 0 0 4.9 2.12 1.95 9.35 25.8 7.14 0];
D1=zeros(2);
```

Starting Points:

```
General definition of the lmi problem
 function [Xo,Yo,gamma]=init1(sys,k,gu)
[a1,b1,c1,d1]=unpck(sys);
%[a1,b1,c1,d1]=unpck(G);
n= size(a1,1); % number of states
u= size(b1,2); % number of inputs
y= size(c1,1); % number of outputs
G=ss(a1,b1,c1,d1);
%hankelsv(G)
%GRED = reduce(G,k);
[sysout,sv] = sysbal(sys);
GRED = hankmr(sysout,sv,k);
[ar,br,cr,dr]=unpck(GRED);

AA=[a1 zeros(n,k) ; zeros(k,n) ar];
BB=[b1;br];
CC=[c1  -cr];
DD=d1;
gg=ss(AA,BB,CC,DD);

gamma=norm(gg,inf);
[A2,B2,C2,D2]=unpck(sysout);
sys1=ss(A2,B2,C2,D2);

co = gram(sys1,'c');
oo = gram(sys1,'o');
Xo=(gu)*inv(co);
Yo=co/(gu);
```

General Definition of the LMI Problem

```
% deflmi.m
function [lmis]=deflmi(G,gamma)
% Getting back system information
 %======================
% SET THE LMI PROBLEM
%======================
%Defining the LMI variables
[A,B,C,D] = unpck(G);
n= size(A,1); % number of states
setlmis([]);
idX=lmivar(1,[n 1]);
idY=lmivar(1,[n 1]);
%Defining the LMI's
[A,B,C,D] = unpck(G);
%First set of LMIs
T=1;
lmiterm([T 1 1 idX],A,1,'s'); % LMI : A*X+X*A'
lmiterm([T 1 1 0],B*B'); % LMI : B*B'
 % Second set of LMIs
lmiterm([T 1 1 idY],1,A,'s'); % LMI : Y*A+A'*Y
lmiterm([T 1 1 0],C'*C); % LMI : C'*C
```

```
% Last LMI
lmiterm([-T 1 1 idX],1,1); % LMI : X
lmiterm([-T 2 1 0],gamma); % LMI : gama*I
lmiterm([-T 2 2 idY],1,1); % LMI : Y
%System definition
lmis=getlmis;
%===================================
% SOLVE LMI problem including rank condition
% by Alternating projection  method
%===================================
%Finding Xo and Yo
% h= decnbr(lmis);
% [tmin,xfeas]=feasp(lmis);
% Xo = dec2mat(lmis,xfeas,idX);
% Yo = dec2mat(lmis,xfeas,idY);
```

Alternating Projection Algorithm:

```
function [Xop, Yop,flag1]= truncation(Xin,Yin,G,gama,k)
temp1=inv(Xin)+Yin;
temp2=Yin-inv(Xin);
Km=redrank(temp2,k); % See file redrank.m
Xou=inv((temp1-Km)/2);
You=(temp1+Km)/2;
n=size(Xou,1);
delta=-1e-10;
flagx=0;
flagy=0;
%%%%%%%%%%%%
[A,B,C,D]=unpck(G);
n= size(A,1); % number of states
% Verifying LMI solution
eigtemp = 0;
eigtempy = 0;
temp=A*Xou+Xou*A'+(B*B');
eigtemp = max(eig(temp));

temp1=A'*You+You*A+(C'*C);
 eigtempy= max(eig(temp1));
if (eigtempy < delta)
      Yop = Yin;
     flagy=1;
     % break
     %return
end % Taking decision according to the previous step
if (eigtemp < delta)
     Xop = Xin;
     flagx=1;
     %break
end
if (flagy & flagx)
    flag=1
   return
else
W=Xin;
V=Yin;
```

```
% Initial values for the loop
[A,B,C,D] = unpck(G);
n= size(A,1); % number of states

delta=-1e-10;

R=eye(n);
DD=eye(n);
setlmis([]);
idX=lmivar(1,[n 1]);
idY=lmivar(1,[n 1]);
idT=lmivar(1,[n 1]);
idS=lmivar(1,[n 1]);

%Defining the LMI's
[A,B,C,D] = unpck(G);
%First set of LMIs
T=1;
lmiterm([T 1 1 idX],A,1,'s'); % LMI : A*X+X*A'
lmiterm([T 1 1 0],B*B'); % LMI : B*B'
 lmiterm([-T 1 1 0],delta);

% Second set of LMIs
lmiterm([T 1 1 idY],1,A,'s'); % LMI : Y*A+A'*Y
lmiterm([T 1 1 0],C'*C); % LMI : C'*C
 lmiterm([-T 1 1 0],delta);
% thirs LMI
lmiterm([-T 1 1 idX],1,1); % LMI : X
lmiterm([-T 2 1 0],gama); % LMI : gama*I
lmiterm([-T 2 2 idY],1,1); % LMI : Y
% Forth LMI
lmiterm([-T 1 1 idT],1,1); % LMI : T
lmiterm([-T 2 1 idX],1,1); % LMI : X
lmiterm([-T 2 1 0],-Xou); % LMI : -Xo
lmiterm([-T 2 2 0],1); % LMI : I
% Last LMI
lmiterm([-T 1 1 idS],1,1); % LMI : S
lmiterm([-T 2 1 idY],1,1); % LMI : Y
lmiterm([-T 2 1 0],-You); % LMI : Yo
lmiterm([-T 2 2 0],1); % LMI : I

%System definition
lmis=getlmis;
h= decnbr(lmis)
Accuracy=1;
Conv=1;
Crt=0;
Ecov=1e-2; % Epsilon tolerance for slow convergence
% (less than 1% in variation)
% Initial values for the mincx funtions
xinit=zeros(h,1);
while Conv > Ecov % Slow convergence criteria
Crtold=Crt;
%---------------------FOR LOOP------------------------------
c=zeros(h,1);
for j=1:h % Inner loop to determine vector c
```

```
[Tj,Sj]=defcx(lmis,j,idT,idS);
c(j)=trace(Tj+Sj);
%c(j) = sum(diag(Tj))+sum(diag(Sj));
end
%-----------------------------------------------------------

options=[10e-5,0,0,0,1];
[copt,Xopt] = mincx(lmis,c,options,xinit,[]);
xinit = mat2dec(lmis,W,V,DD,R); % Initial guess for the next
iteration
Xk=dec2mat(lmis,Xopt,idX); % Building matrix Xn from decision vector
Yk=dec2mat(lmis,Xopt,idY); % Building matrix Yn from decision vector
Sk=dec2mat(lmis,Xopt,idS); % Building matrix Xn from decision vector
Tk=dec2mat(lmis,Xopt,idT); % Building matrix Yn from decision vector

Crt= trace(Sk+Tk);
Conv=abs((Crt-Crtold)/Crt);
W=Xk;

%Yn=Yk;
V=redrank(Yk,k);

%V=Yn;
end
%-----------------------------------------------------------
Xop=W; % Displaying results
Yop=V;
flag1=flagx+flagy;
end
```

Model Reconstruction:

```
% LINEAR MODEL REDUCTION
% modrec.m
function [Gr]=modrec(G,Xoptimal,Yoptimal,k,g)
% %
 [A,B,C,D]=unpck(G);
n= size(A,1); % number of states
p=size(C,1);
m=size(B,2);

%Building matrix J
J=Yoptimal-g*g*inv(Xoptimal);
[u,s,v]=svd(J);
Lx=u(:,1:k)*sqrt(s(1:k,1:k)); % N*N' = Yopt-inv(Xopt)
% Initialize loop variables
% Gr=[];
%          L=A'*Yoptimal+Yoptimal*A;
%          h=inv(N'*inv(L)*N); %dummy variable
%          Ar=h*(eye(k)-N'*inv(L)*A'*N);
%          Br=-h*N'*inv(L)*Yoptimal*B;
%          Cr=-C*inv(L)*A'*N-C*inv(L)*N*Ar;
%          Dr=D-C*inv(L)*(eye(n)-
N*inv(N'*inv(L)*N)*N'*inv(L))*Yoptimal*B;
%          Gr=pck(Ar,Br,Cr,Dr);
```

```
 Gr=[];
setlmis([]);
L = lmivar(2,[p+k m+k]);
lmiterm([-1 1 1 0],1);
lmiterm([-1 1 2 L],1,1);
lmiterm([-1 2 2 0],1);

lmiterm([-2 1 1 L],1,1);

L_lmi=getlmis;
[topt,xopt] = feasp(L_lmi);
l = dec2mat(L_lmi,xopt,L)
Rx=inv(Lx'*inv(L)*Lx); %dummy variable

M1=[zeros(p,m) zeros(p,k);
    zeros(k,m)   Rx*Rx];
Q22=[A*Xoptimal+Xoptimal*A'  B; B'  -eye(m)];
Q11=[-g*g*eye(p) C*Lx*Rx;Rx*Lx'*C'   zeros(k)];
Q12=[C*Xoptimal    D; Rx*Lx'*A'     zeros(k,p)];
M2=[zeros(m,n) eye(m); Rx*Lx' zeros(k,m)];
G1=(M1-Q12*inv(Q22)*M2')*inv(M2*inv(Q22)*M2');
G3=sqrt(M2*inv(Q22)*M2');
G2=(-Q11+Q12*inv(Q22)*Q12'-G1*G3*G3*G1');
G=G1+G2*L*G3
Dr=zeros(p,m);
 for i=1:p
     for j=1:m
         Dr(i,j)=G(i,j);
     end
 end

 Cr=zeros(p,k);
 for i=1:p
     for j=m+1:m+k
         Cr(i,j-m)=G(i,j);
     end
 end
 Br=zeros(k,m);
 for i=p+1:p+k
     for j=1:m
         Br(i-p,j)=G(i,j);
     end
 end
Ar=zeros(k,k);

  for i=p+1:p+k
     for j=m+1:m+k
         Ar(i-p,j-m)=G(i,j);
     end
  end
 Gr=pck(Ar,Br,Cr,Dr);
```

H_{FH} Model Reduction Using CCA:

Main file

```
% System definition file
% Input the system
clear all;
Erof=1e-3;
clc;
time=clock;
%System matrices definition
system;
packfile; % See file packfile.m
% Desired order of the reduced system
k=5;
% Upper and lower boundaries for gamma optimal
[gl,gu]= SDB(G,VSinf,k); % See file SDB.m
flag=0;
flag2=0;
itr=0;
itr_d=[];
itr_dd=[];
while flag==0
tol=(gu-gl)/gl;
if (tol <= 0.01) & flag2==1
gama
flag = 1;
else
gama=(gu+gl)/2;
[lmis,Xo,Yo]=deflmi(G,VSinf,gama); % See file setlmi.m
itr=itr+1;
for i=1:10
[Xin,Yin,itr_d]=cca(lmis,Xo,Yo,gama,k,n,itr_d); % See file cca.m
Km=[Xin eye(n);
eye(n) Yin];
nr=rank(Km,Erof)-n
if nr <= k
Xop=Xin;
Yop=Yin;
flag1=1;
break
else
Xo=Xin;
Yo=Yin;
flag1=0;
end
% end
end
if flag1==1
gu=gama
flag2=1
else
gl=gama
flag2=0
end
end
end
 [Gr]= modrec(G,VSinf,Xop,Yop,k); % See file modrec.m
```

```
time=etime(clock,time);
clc;
l= size(itr_d,1);
mm =1:1:l;
figure(3);
plot(mm,itr_d(:,2));
grid on;
xlabel('Gamma Convergence','FontSize',14);
ylabel('Iteration','FontSize',14);
title(['Gamma Convergence when K=',num2str(k)],'FontSize',14);
time
gama
```

==========

System

```
% System definition file AUTM system
A1 = [0 1 0 0 0 0 0 0 0 0 0 0 ;
    -0.202 -1.15 0 0 0 0 0 0 0 0 0 0 ;
    0 0 0 1 0 0 0 0 0 0 0 0 ;
    0 0 0 0 1 0 0 0 0 0 0 0 ;
     0 0 -2.36 -13.6 -12.8 0 0 0 0 0 0 0 ;
    0 0 0 0 0 0 1 0 0 0 0 0 ;
    0 0 0 0 0 0 0 1 0 0 0 0 ;
    0 0 0 0 0 -1.62 -9.4 -9.15 0 0 0 0 ;
    0 0 0 0 0 0 0 0 0 1 0 0 ;
    0 0 0 0 0 0 0 0 0 0 1 0 ;
    0 0 0 0 0 0 0 0 0 0 0 1;
    0 0 0 0 0 0 0 0 -188 -111.6 -116.4 -20.8];


B1= [ 0  0; 4.1486 1.0439; 0 0; 0 0; 2.6775  -1.794; 0  0; 0  0;
4.1486 1.0439 ;0  0; 0  0; 0  0; 2.6775 -1.794];

C1= [0.2640 0.8060 -1.420 -15.0 0 0 0 0 0 0 0 0;
    0 0 0 0 0 4.9 2.12 1.95 9.35 25.8 7.14 0];
D1=zeros(2);

Ac= [A1];
Bc= [B1];
Cc= [C1];
Dc= [D1];
```

Lower and Upper Boundaries

```
% SDB.m
% The function bounds provides upper (ub) and
% lower (lb) boundaries for the calculation of
% gamma optimal, given the desired order of the
% reduced system.
function [gl,gu]= SDB (G,VSinf,k)
% Retrieving system information
VA=VSinf(1);
VB=VSinf(2);
VC=VSinf(3);
```

```
VD=VSinf(4);
NV=VSinf(5);
n=VSinf(6);
% Creating the system matrices
hk=[];
for r=1:VSinf(5)
Gi=xtracti(G,r,1);
[du,hki]=sysbal(Gi);
hk=[hk hki(k+1)]; % Concatenation of the Hankel
% singular values of each vertix
end
% Calculating the lower bound
gl=max(hk); % Gamma lower bound
% Calculating the upper bound
[vma,vmb,vmc,vmd]= vm(G,VSinf); % See file vm.m
%Defining the LMI variables
setlmis([]);
P=lmivar(1,[n 1]);
Q=lmivar(1,[n 1]);
%Defining the LMI's
T=1; % Number of the lmi
for i=1:VA
for j=1:VB

Ai=xtracti(vma,i,1);
Bi=xtracti(vmb,j,1);
%First set of LMIs
lmiterm([T 1 1 P],Ai,1,'s'); % LMI : A*P+P*A'
lmiterm([T 1 1 0],Bi*Bi'); % LMI : B*B'
T=T+1;
end
end
for i=1:VA
for p=1:VC
Ai=xtracti(vma,i,1);
Ci=xtracti(vmc,p,1);
% Second set of LMIs
lmiterm([ T 1 1 Q],Ai',1,'s'); % LMI : Q*A+A'*Q
lmiterm([ T 1 1 0],Ci'*Ci); % LMI : C'*C
T=T+1;
end
end
lmiterm([-T 1 1 P],1,1); % Third LMI
lmiterm([-T 2 2 Q],1,1);
upperb=getlmis;
% Starting minimization of the gamma upper limit
h= decnbr(upperb);
c=zeros(h,1);
%---------------------FOR LOOP-----------------------------------
 for j=1:h, % Inner loop to determine vector c
[Pj,Qj]=defcx(upperb,j,P,Q);
c(j)=trace(Pj+Qj);
end
%------------------------------------------------------------------
```

```
options=[1e-5,0,0,0,1];
[copt,Xopt] = mincx(upperb,c,options);
Popt=dec2mat(upperb,Xopt,P);
Qopt=dec2mat(upperb,Xopt,Q);
R=chol(Popt); % R'*R = P
[u,s,v]=svd(R*Qopt*R');
T=inv(R'*u*inv(sqrt(sqrt(s))));
sigma=T*Popt*T';
temp=diag(sigma);
gu=2*sum(temp(k+1:length(temp))); % Gamma upper bound
```

General definition of the LMI problem

```
% deflmi.m
% General definition of the lmi problem
function [lmis,Xo,Yo]=deflmi(G,VSinf,gama)
% Getting back system information
VA=VSinf(1);
VB=VSinf(2);
VC=VSinf(3);
VD=VSinf(4);
NV=VSinf(5);
n=VSinf(6);
[vma,vmb,vmc,vmd]= vm(G,VSinf);
%======================
% SET THE LMI PROBLEM
%======================
%Defining the LMI variables
setlmis([]);
idX=lmivar(1,[n 1]);
idY=lmivar(1,[n 1]);
%Defining the LMI's
T=1; % Number of the lmi
lmiterm([-T 1 1 idX],1,1); % LMI : X>0
T=T+1
lmiterm([-T 1 1 idX],1,1); % LMI : X>0
T=T+1
for i=1:VA
for j=1:VB
Ai=xtracti(vma,i,1);
Bi=xtracti(vmb,j,1);
%second set of LMIs
lmiterm([T 1 1 idX],Ai,1,'s'); % LMI : A*X+X*A'
lmiterm([T 1 1 0],Bi*Bi'/gama); % LMI : B*B'/gama
T=T+1;
end
end
for i=1:VA
for p=1:VC
Ai=xtracti(vma,i,1);
Ci=xtracti(vmc,p,1);
% third set of LMIs
lmiterm([T 1 1 idY],1,Ai,'s'); % LMI : Y*A+A'*Y
lmiterm([T 1 1 0],Ci'*Ci/gama); % LMI : C'*C/gama
T=T+1;
end
end
```

```
% Last LMI
%lmiterm([-T 1 1 idX],1,1); % LMI : X
%lmiterm([-T 2 1 0],1); % LMI : 1
%lmiterm([-T 2 2 idY],1,1); % LMI : Y
%System definition
%System definition
lmis=getlmis;
%===================================
% SOLVE LMI problem including rank condition
% by cone complementarily method
%===================================
%Finding Xo and Yo
h= decnbr(lmis);
[tmin,xfeas]=feasp(lmis);
Xo = dec2mat(lmis,xfeas,idX);
Yo = dec2mat(lmis,xfeas,idY);
```

CONE COMPLEMENTARITY ALGORITHM

```
% cca.m
% CONE COMPLEMENTARITY ALGORITHM
function [Xin,Yin,itrd]=cca(lmis,Xo,Yo,gama,k,n,itr_d)
W=Xo;
V=Yo;
m=size(itr_d,1);
setlmis([]);
idX=lmivar(1,[n 1]);
idY=lmivar(1,[n 1]);
h= decnbr(lmis);
Accuracy=1;
Conv=1;
Crt=0;
Ecov=1e-2; % Epsilon tolerance for slow convergence
% (less than 1% in variation)
% Initial values for the mincx funtions
xinit=zeros(h,1);
while Conv > Ecov % Slow convergence criteria
Crtold=Crt;
m=m+1;
itr_di=[m  gama];
itrd=[itr_d ;itr_di];

%---------------------FOR LOOP-----------------------------
c=zeros(h,1);
for j=1:h % Inner loop to determine vector c
[Xj,Yj]=defcx(lmis,j,idX,idY);
c(j)=trace(V*Xj+W*Yj);
end
%------------------------------------------------------------
options=[10e-5,0,0,0,1];
[copt,Xopt] = mincx(lmis,c,options,xinit,[]);
xinit = mat2dec(lmis,W,V); % Initial guess for the next iteration
Xk=dec2mat(lmis,Xopt,idX); % Building matrix Xn from decision vector
Yk=dec2mat(lmis,Xopt,idY); % Building matrix Yn from decision vector
Crt= trace(V*Xk+W*Yk);
Conv=abs((Crt-Crtold)/Crt);
```

```
W=Xk;
V=Yk;
end
%-------------------------------------------------------------
Xin=W; % Displaying results
Yin=V;

% modrec.m
% model reconstruction code
function [Gr]=modrec(G,VSinf,Xoptimal,Yoptimal,k)
% Getting back system information
VA=VSinf(1);
VB=VSinf(2);
VC=VSinf(3);
VD=VSinf(4);
NV=VSinf(5);
n=VSinf(6);
[vma,vmb,vmc,vmd]= vm(G,VSinf);
%Building matrix J
J=Yoptimal-inv(Xoptimal);
[u,s,v]=svd(J);
N=u(:,1:k)*sqrt(s(1:k,1:k)); % N*N' = Yopt-inv(Xopt)
% Initialize loop variables
Gr=[];
%for i=1:NV             %%%seeeee
%ivv =[ ivv ; i];         %%
%end                    %%%

% Main loop
for i=1:VA
  for j=1:VB
    for p=1:VC
      for q=1:VD
        Ai=xtracti(vma,i,1);
        Bi=xtracti(vmb,j,1);
        Ci=xtracti(vmc,p,1);
        Di=xtracti(vmd,q,1);
        Li=Ai'*Yoptimal+Yoptimal*Ai;
        h=inv(N'*inv(Li)*N); %dummy variable
        Ari=h*(eye(k)-N'*inv(Li)*Ai'*N);
        Bri=-h*N'*inv(Li)*Yoptimal*Bi;
        Cri=-Ci*inv(Li)*Ai'*N-Ci*inv(Li)*N*Ari;
        Dri=Di-Ci*inv(Li)*(eye(n)-
N*inv(N'*inv(Li)*N)*N'*inv(Li))*Yoptimal*Bi;
        Gri=pck(Ari,Bri,Cri,Dri);
        Gr=[ Gr; Gri];
      end
    end
   end
end

Gr= vpck(Gr,[1:NV]);
```

Orthogonal projection of a symmetric matrix:

```
% redrank.m
% NONCONVEX SET for Alternating projection method
% Orthogonal projection of a symmetric matrix to a
% given order NR symmetric matrix.
%
function zout = redrank(zin,nr)
[nx,dum] = size(zin);
[u,d,v] = svd(zin);
for i = nr+1:nx
d(i,i) = 0.0;
end
zout = u*d*v';
zout = (zout+zout')/2;
```

This file used for uncertain system:

```
% Obtain information from the system such as number
% of inputs, outputs, states and
% pack the matrices in a varying matrix format
% Getting system information
n= size(A1,1); % number of states
u= size(B1,2); % number of inputs
y= size(C1,1); % number of outputs
% Getting the number of vertices of each matrix
VA= size (Ac,1)/n; % # of vertices of A
VB= size (Bc,1)/n; % # of vertices of B
VC= size (Cc,1)/y; % # of vertices of C
VD= size (Dc,1)/y; % # of vertices of D
NV=VA*VB*VC*VD; % Total number of vertices
VSinf=[ VA VB VC VD NV n u y];
% Creating the varying matrices
iv=[];
for i=1:NV
iv=[ iv ; i];
end
vma= vpck(Ac,iv(1:VA,1));
vmb= vpck(Bc,iv(1:VB,1));
vmc= vpck(Cc,iv(1:VC,1));
vmd= vpck(Dc,iv(1:VD,1));
G=[];
for i=1:VA
for j=1:VB
for p=1:VC
for q=1:VD
Ai=xtracti(vma,i,1);
Bi=xtracti(vmb,j,1);
Ci=xtracti(vmc,p,1);
Di=xtracti(vmd,q,1);
Gi=pck(Ai,Bi,Ci,Di);
G=[G ; Gi];
end
end
end
end
G=vpck(G,iv(1:NV,1));
```

```
clear Gi Ai Bi Ci Di i j p q;
================================================================
```

Alternating Projection Algorithm:

```
% trunc.m
function [Xop, Yop,flag1]= trunc(Xin, Yin,G,gama,VSinf,k)
temp1=inv(Xin)+Yin;
temp2=Yin-inv(Xin);
Km=redrank(temp2,k); % See file redrank.m
Xou=inv((temp1-Km)/2);
You=(temp1+Km)/2;
n=size(Xou,1);
delta=-1e-10;
flagx=0;
flagy=0;
[Xop,flagx]= xproj(Xou,G,VSinf,gama,delta); % See file xproj.m
[Yop,flagy]= yproj(You,G,VSinf,gama,delta); % See file yproj.m
flagx;
flagy;
flag1=flagx+flagy;

% xproj.m
function [Xop,flagx]= xproj(Xou,G,VSinf,gama,delta)
VA=VSinf(1);
VB=VSinf(2);
NV=VSinf(5);
n=VSinf(6);
[vma,vmb,vmc,vmd]= vm(G,VSinf);
% Verifying LMI solution
eigtemp = zeros(VA*VB,1);
t=1;
for i=1:VA
for j=1:VB
a=xtracti(vma,i,1);
b=xtracti(vmb,j,1);
temp=a*Xou+Xou*a'+(b*b')/gama;
eigtemp(t,1) = max(eig(temp));
t=t+1;
end
end
% Taking decision according to the previous step
if (max(eigtemp) < delta)
     Xop = Xou;
     flagx=1;
    % break
    return
else
%Defining the LMI variables
setlmis([]);
idX=lmivar(1,[n 1]);
idZ=lmivar(1,[n 1]);
%Defining the LMI's

T=1; % Number of the lmi
for i=1:VA
  for j=1:VB
     Ai=xtracti(vma,i,1);
```

```
     Bi=xtracti(vmb,j,1);
     %First set of LMIs
     lmiterm([T 1 1 idX],Ai,1,'s'); % LMI : A*X+X*A'
     lmiterm([T 1 1 0],Bi*Bi'/gama); % LMI : B*B'/gama
     lmiterm([-T 1 1 0],delta);
     T=T+1;
   end
end
    lmiterm([T 1 1 idZ],-1,1);
    lmiterm([T 2 1 idX],-1,1);
    lmiterm([T 2 1 0],Xou);
    lmiterm([T 2 2 0],-1);
   lmis = getlmis;
   nvar = decnbr(lmis);
   cvec = zeros(nvar,1);
   for j = 1:nvar
      vz = defcx(lmis,j,idZ);
      cvec(j,1) = trace(vz);
   end
  [copt,xopt] = mincx(lmis,cvec,[1e-2 100 -1 0 1]);
  Xop = dec2mat(lmis,xopt,idX);
  flagx=0;
end


% yproj.m
function [Yop,flagy]= yproj(You,G,VSinf,gama,delta)
VA=VSinf(1);
VB=VSinf(2);
VC=VSinf(3);
NV=VSinf(5);
n=VSinf(6);
[vma,vmb,vmc,vmd]= vm(G,VSinf);

% Verifying LMI solution
eigtemp = zeros(VA*VC,1);
t=1;
for i=1:VA
for j=1:VC
a=xtracti(vma,i,1);
c=xtracti(vmc,j,1);
temp=a'*You+You*a+(c'*c)/gama;
eigtemp(t,1) = max(eig(temp));
t=t+1;
end

end


% Taking decision according to the previous step

if (max(eigtemp) < delta)
     Yop = You;
     flagy=1;
    % break
```

```
Return
Else
%Defining the LMI variables
setlmis([]);
idY=lmivar(1,[n 1]);
idZ=lmivar(1,[n 1]);
%Defining the LMI's

T=1; % Number of the lmi
for i=1:VA
  for j=1:VC
     Ai=xtracti(vma,i,1);
     Ci=xtracti(vmc,j,1);
     %First set of LMIs
     lmiterm([T 1 1 idY],1,Ai,'s'); % LMI : A*X+X*A'
     lmiterm([T 1 1 0],Ci'*Ci/gama); % LMI : B*B'/gama
     lmiterm([-T 1 1 0],delta);
     T=T+1;
   end
end
    lmiterm([T 1 1 idZ],-1,1);
    lmiterm([T 2 1 idY],-1,1);
    lmiterm([T 2 1 0],You);
    lmiterm([T 2 2 0],-1);
   lmis = getlmis;
   nvar = decnbr(lmis);
   cvec = zeros(nvar,1);
   for j = 1:nvar
      vz = defcx(lmis,j,idZ);
      cvec(j,1) = trace(vz);
   end
  [copt,yopt] = mincx(lmis,cvec,[1e-2 100 -1 0 1]);
  Yop = dec2mat(lmis,yopt,idY);
  flagy=0;
end
```