The Pennsylvania State University

The Graduate School

College of Engineering

# DIMENSIONAL TRANFORMATION: A NOVEL METHOD FOR GAIN-SCHEDULING AND ROBUST CONTROL

A Thesis in

Mechanical Engineering

by

Haftay Hailu



Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

May 2006

The thesis of Haftay Hailu was reviewed and approved* by the following:

Sean Brennan
Assistant Professor of Mechanical Engineering
Thesis Advisor
Chair of Committee

Chris Rahn
Professor of Mechanical Engineering

Mario Sznaier
Professor of Electrical Engineering

Kon-Well Wang
William E. Diefenderfer Chaired Professor in Mechanical Engineering.

H. Joseph Sommer III
Professor and Interim Head
Department of Mechanical and Nuclear Engineering

*Signatures are on file in the Graduate School

## ABSTRACT

This thesis focuses on developing a technique of dimensional transformation to solve advanced controller design problems, specifically gain-scheduling and robust control methods. The developed technique reformulates the system representation in preferential dimensionless form that is more tenable for gain scheduling and robust control designs. The current work shows that the dimensionless formulation gives advantage in terms of reducing complexity and conservativeness of the control synthesis as compared to the dimensional form.

The complexity of a gain scheduling control design increases exponentially with the number of scheduling parameters. This thesis presents a method called dimensional transformation that reduces the number of scheduling parameters by reformulating the dynamic representation in dimensionless form. The choice of dimensionless description is usually preferred because any transformation to dimensionless representation is guaranteed to reduce variable dependence: the number of dimensionless parameters is always less than or equal to that of the classical representation (a result of the classic Pi-theorem). However, dimensional transformations are not unique. Some transformations – while reducing the total number of system parameters, scheduled or unscheduled – may have a negative effect on a gain-scheduling control algorithm because they may inadvertently increase the number of scheduling parameters. This work explores in detail conditions necessary such that dimensional transformation are guaranteed to present the minimum number of gain scheduling parameters for a control system design.

The same principle of dimensional transformation is explored as a method to reduce the size of parametric uncertainty block in robust controller synthesis. This simplification is performed using the dimensional transformation by appropriate matrices followed by LFT reformulation in the dimensionless domain. This reduces the conservativeness of the robust control synthesis. For example, in the $\mu$-synthesis framework, if the uncertainty block size is greater than three, then only the upper bound can be computed, and this upper bound can be arbitrarily larger than the actual structured singular value resulting in a more conservative controller synthesis. Through the method of dimensional transformation, the size of a given uncertainty block can be reduced by up to three or more dimensions. Depending on the problem, this might allow for current techniques of robust control synthesis to be extended into significant new problems.

This thesis also discusses methods for a robust simultaneous control technique for systems whose system parameters are inherently coupled. The current work shows the potential of the proposed method for designing a unified robust controller that can be implemented through parametric adaptation. The goal is to obtain a robust, adaptive and modular controller for a group of systems. When considering collective group of systems, passenger vehicles for example, coupling is present due to optimization inherent in vehicles and engineering and natural systems. When such general coupling exists, the systems as a group can be represented in a more dense collection that gives advantage for robust controller synthesis. This approach is tested using a problem focusing on the lateral control of a scaled-vehicle-system on a rolling-roadway simulator.

# TABLE OF CONTENTS

## LIST OF FIGURES

## LIST OF TABLES

## ACKNOWLEDGEMENTS

I owe a debt of gratitude to many people: family, professors, fellow students and friends who have, directly or indirectly, contributed to the completion of this thesis. First of all, I would like to express my gratitude for my advisor Dr. Sean Brennan for his advice and enthusiasm. I was fortunate to have your guidance throughout this research. I also would like to forward my special thanks to my advisory committee, whom I had the honor to have them in my advisors committee and also for teaching me great lessons throughout my school at Penn State. The members of this great team include: Dr. Mario Sznaier, Dr. Bohdan T. Kulakowski, Dr. Chris Rahn and Dr. Kon-Well Wang. In this final version, I like to remember Dr. Bohdan Kulakowski who passed away five days after attending the defense of this thesis due to a terrible accident. My sympathy goes to his family and may God bless his soul.

My special thanks also go to the students in our research group, The Brennan Research Group. I specially like to thank Vishisht Gupta, without whom the implementation of this work would have been very difficult. Thanks for developing and maintaining the scaled-vehicle and the rolling-roadway simulator in good condition. I also would like to extend my thanks to Sittikorn Lapapong for collecting and updating the vehicle parameters database.

I also would like to extend my heart felt thanks to my family for their love and endless support. Special thanks also goes to all my friends whose encouragement has helped me cope with the challenges in this long journey.

## Chapter 1

# Introduction

This thesis focuses on the use of dimensional transformation in advanced controller design methods, specifically with gain scheduling and robust control methods. The term "dimensional transformation" is used hereafter to refer to a method of transforming a dynamic system model expressed in a given dimensional basis units (or simply, units of measurement) into another model expressed in different units of measurement. These transformations include those that produce models that have "dimensionless" units, a very special category of system representations that will be given a large focus in this work. The equivalence between the two system models with different unit systems is established based on the equivalence of relative sensitivity. This includes all measures of sensitivity familiar to a controls engineer: parametric sensitivity, sensitivity to disturbances, etc. This thesis focuses primarily on parametric sensitivity, examining unit transformations that provide benefit to the task of developing a gain-scheduled and/or robust controller.

## 1.1 Motivation

The main motivation of the thesis is to develop a technique to reduce complexity and conservativeness in two of the most important advanced controller synthesis methods, namely gain scheduling and robust control. This thesis shows, using many examples, why the unit-transformation technique is important in control problems. These control problems that this thesis focuses are categorized into the following three major categories: 1) gain-scheduling problem, 2) robust control problem, and 3) simultaneous robust control of a family of systems through parametric adaptation. A detailed description of each is given below.

### 1.1.1 Gain Scheduling Problem

The use of gain scheduling as a controller design method has been very useful in modifying the behavior of nonlinear systems and linear parameter varying (LPV) systems. However, there are practical and theoretical limitations with current design methods with respect to the number of scheduling parameters. In gain scheduling, as the number of varying parameters increases, so does the complexity of the controller synthesis. For example, in the case of robust gain scheduling of a system with non-affine system-parameter dependence, using the LPV-LMI framework, gain-scheduling is practically limited to only 3 or 4 scheduling parameters. Such limitation is due to the computational complexity involved in solving the LMI feasibility problems that are currently performed using the method of gridding [1-6], i.e., picking many points along the path of parametric variation of each parameter. Even for cases where affine system-parameter dependence is assumed, control synthesis requires solving $2^N$ LMI feasibility problems for $N$ - scheduling parameters. The affine system-parameter dependence is not a generalized assumption applicable to most systems: many practical problems are considered affine only with the time rate of change of the parameters, not affine in the parameters themselves. When the system is affine with respect to the rates of changes and non-affine with respect to the parameters, the feasibility problem is solved at the end points of the affine parameters. For the non-affine parameters, the gridding method is used at great computational cost. Any means to reduce this cost could have great benefit to controller synthesis techniques applied to practical problems.

This thesis develops a technique for reducing the number of gain scheduling parameters using unit transformations. The result is a simpler scheduling parametric space. The developed technique of parametric space reduction has two parts: first a mathematical rule developed to quantify the reduction of the parameters in a given dimensionless representation. The second part is an optimization algorithm that searches through dimensionless transformations to find the minimum number of scheduling parameters of the system over all possible transformations.

### 1.1.2 Robust Control Problem

The method of robust control synthesis has been among the most effective modern control design methods for systems that exhibit some uncertainty in their models. For robust controller synthesis using current tools of $H_\infty$ and $\mu$-synthesis, the plant and uncertainty must be formulated in linear fractional transformation (LFT) form with the uncertainty separated from the nominal model and represented as a separate block. However, the order and size of the uncertainty block (number of individual uncertainty blocks in the whole uncertainty block) of the LFT form affects the complexity and conservativeness of the control synthesis [7-10]. The method developed in this thesis reduces the size of the uncertainty block using dimensional transformation. A side-by-side numerical comparison of a dimensional and dimensionless control synthesis is performed to illustrate the advantages of the current technique. This thesis shows, using the return difference matrix determinant condition for robust stability that the dimensionless approach allows larger amounts of parametric perturbation compared to the dimensional approach. Three main drawbacks of the proposed method are presented, along with problem-dependent causes and possible remedies.

### 1.1.3 The Vehicle Autopilot: A Simultaneous Robust Control through Parametric Adaptation

Man-made or natural systems have parameters that are generally coupled due to design constraints or optimization, i.e., the values of the system parameters either increase/decrease as a group. An example would be passenger vehicles whose parameters and behavior collectively change as the vehicles size increases/decreases. The range in variation can be dramatic: passenger vehicles, for example, have different sizes and span from the smallest compact car to the biggest sport utility vehicles (SUVs). Despite such variation, parameter variations among these systems tend to vary in lockstep relationship across size: either all increasing or all decreasing together. For instance, as the size of a vehicle increases, so does the length, mass and mass moment of inertia. These trends exist in addition to the mathematical relationship expressed by a vehicle’s dynamic model yet are often excluded from consideration when performing robust controller synthesis.

This thesis shows how these proportionality trends in system parameters can be exploited to establish a common model representative of all vehicles, a model with a very compact level of vehicle-to-vehicle variation or uncertainty. This type of collective modeling is very amendable for developing a single, unified, robust controller synthesis that generalizes to the entire family of plants in the domain. The proposed technique also shows how to reverse-transform the unified controller for implementation on the individual systems of the family. The generalization and specificity transformations are performed though parametric adaptation, and hence saves energy and money spent on tuning a new controller for every vehicle model in a production line. The method discussed in this thesis can also be applied to vehicles currently in development and production that share similar body styles and component lines, but whose parametric differences are too large to allow a classical controller to be transferred from one model to another.

## 1.2 Main Contribution of this Thesis

Previous work in the area of applying dimensional analysis in control [11-15], especially the contribution of Brennan [12] in this area is significant. This thesis is different from previous work mainly in the following respects:

1. This thesis develops, for the first time, the reduction of parametric space for gain scheduling using the dimensional transformation method with a formal rule to determine the bounds of reduction. The thesis also presents an optimization algorithm to search for the minimum number of scheduling parameters of a given gain scheduling problem.

2. This thesis develops a methodology for the reduction of parametric uncertainty block size using dimensional transformation. The advantages of this reduction are the potential to reduce conservativeness and complexity in the robust controller synthesis.
3. A side-by-side comparison of dimensional and dimensionless robust controllers is performed. This is perhaps the first direct comparison to show the advantages of the technique discussed in this thesis.
4. The application of dimensional analysis for a unified robust control design for a group passenger vehicles has been studied in Brennan [14] based on the stacked sensitivity approach and by representing the plant variation as a frequency-domain dynamic uncertainty in input-output model. The approach in the current thesis is from a parametric uncertainty perspective. The goal of including parametric structure is to allow for a less-conservative control synthesis. Both $H_\infty$ and $\mu$-synthesis/analysis tools are used herein. The parametric uncertainty approach also results in a lower order controllers compared to the dynamic uncertainty because the dynamic uncertainty approach by Brennan [14] has a frequency dependent uncertainty weights that unnecessarily increase the order of the robust controller.

## 1.3 Organization of the Remaining Chapters

The remainder of this thesis is organized as follows: the second chapter discusses dimensional analysis, a widely used concept in many engineering applications including fluid mechanics, heat transfer, etc. It is also the basis for the dimensional transformation method used throughout this thesis. The history of dimensional analysis as well many applications are briefly considered.

The third chapter introduces the method of dimensional transformation and presents specific applications to dynamics and control problems. The main goal of this chapter is to discuss the dimensional transformation steps by formulating the transformation as a matrix algebra problem.

The fourth chapter presents the application of dimensional transformation for gain scheduling parameter reduction which has a practical importance in many gain-scheduled control design methods including switching controllers and parameterized gain scheduling based on the LPV-LMI framework. More specifically, this approach presents a method of alleviating some of the current practical limitations on the number of varying parameters that are used in gain scheduling. This chapter discusses the development of a gain scheduling parameter reduction rule and an optimization of the gain scheduling parameters over all possible dimensionless representation of a given system.

The fifth chapter discusses how specific unit transformations result in parametric uncertainty block size reduction within a robust control synthesis problem. Using a numerical example, a comparative study of robust control synthesis between the dimensional and dimensionless domains is discussed. Three potential drawbacks of the technique are raised and solutions are proposed.

The sixth chapter discusses the selective transformation of systems whose parameters exhibit a similar trend as the size of the system is increased or decreased. The motivating example is development of a vehicle-general autopilot. It is shown that a compact representation of all vehicles is obtained in just one model by accounting for how the mass, length and velocity scale across size-domains of the vehicle. A method of exploiting such parametric variation is developed via a technique that unifies the robust control design process. This chapter also shows that once the control synthesis is performed in the dimensionless domain it can them be parametrically transformed to the individual vehicles for implementation. The technique developed in this chapter is finally tested using an actual vehicle.

Finally, the seventh chapter summarizes the main results of the thesis. The future directions of research in this area are also discussed.

## 1.4 References


[1] G. Papageorgiou and K. Glover, "Taking robust LPV Control into flight on the VAAC Harrier," the 39th IEEE Conference on Decision and Control, Sydney, Australia, 2000.

[2] S. Ganguli, A. Marcos, and G. J. Balas, "Reconfigurable LPV Control Design for Boeing 747-100/200 Longitudinal Axis," American Control Conference, Anchorage, AK, 2002.

[3] F. Wu, "An Unified Framework for LPV System Analysis and Control Synthesis," the 39th IEEE Conference on Decision and Control, Sydney, Australia, 2000.

[4] P. Apkarian and R. J. Adams, "Advanced Gain-Scheduling Techniques for Uncertain Systems," *IEEE Transactions on Control Sysytems Technology*, vol. 6, pp. 21-32, 1998.

[5] S. J. Fedigan, C. R. Knospe, and R. D. Williams, "Gain-Scheduled Control for Substructure Properties," Proceedings of the 1998 American Control Conference, 1998.

[6] F. Bruzelius, "LPV-Based Gain Scheduling, An $H_\infty$–LMI Approach," in *Signals and Systems*. Göteborg: Chalmers University of Technology, 2002, pp. 112.

[7] G. M. Voorsluijs and J. A. Mulder, "Parametric-Dependent Robust Control for a Rotorcraft UAV," AIAA Guidance, Navigation, and Control Conference and Exhibit, San Francisco, CA, USA, 2005.

[8] M. K. H. Fan, A. L. Tits, and J. C. Doyle, "Robustness in the Presence of Mixed Parametric Uncertainty and Unmodeled Dynamics," *IEEE Transaction on Automatic Control*, vol. 36, pp. 25-38, 1991.

[9] A. Packard, M. K. H. Fan, and J. C. Doyle, "A Power Method for the Structured Singular Value," Proceedings of the 27th IEEE Conference on Decision and Control, Austin, Texas, 1988.

[10] Moulin, "Modeling of Aeroservoelastic Systems with Structural and Aerodynamic Variations," *AIAA Journal*, vol. 43, 2005.

[11] S. Tavakoli and M. Tavakoli, "Optimal Tuning of PID Controllers for First Order plus Time Delay Models Using Dimensional Analysis," The 4th IEEE International Conference on Control and Automation, Canada, 2003.

[12] S. N. Brennan, "On Size and Control: The Use of Dimensional Analysis in Controller Design," in *Mechanical Engineering*. Urbana-Champaign: University of Illinois, 2002, pp. 370.

[13] K. J. Astrom, C. C. Hang, P. Persson, and W. K. Ho, "Towards Intelligent PID Control," *Automatica*, vol. 28, pp. 1-9, 1992.

[14] S. Brennan and A. Alleyne, "Generalized H-infinity Vehicle Control Utilizing Dimensional Analysis," American Control Conference, Denver, CO, 2003.

[15] S. Brennan and A. Alleyne, "H-infinity vehicle control using nondimensional perturbation measures," Proceedings of the American Control Conference, Anchorage, AK, 2002.

# Chapter 2

# Dimensional Analysis

## 2.1 Introduction

This chapter provides an overview of the history of dimensional analysis. It also states the most important statement of dimensional analysis known as the Pi-Theorem and sometimes referred to as the Buckingham Pi-Theorem [1]. An example of a dimensionless representation of a dynamic systems model is also discussed. This thesis presents, in the later chapters, the application of these principles to control theory.

In the engineering domain and the natural sciences "dimensional homogeneity", e.g. units on both sides of an equality or inequality match, is required for all valid mathematical relationships [2]. Indeed all dimensionally inhomogeneous relationships are agreed on to be wrong by definition. This principle provides a restriction on the search space of admissible functional relations for a given problem [3]. Conventional knowledge discovery systems often ignore or even violate this requirement and have therefore to search a much larger search space containing also inadmissible functional relationships.

The idea of a "dimension" had its origins in ancient Greek geometry. It was then considered that lines had one dimension, surfaces had two dimensions and solids had three dimensions (see Martins [3]). These dimensions were related to the rule of principle of homogeneity, according to which only magnitudes of the same kind can be added or equated, and only these had numeric ratio. However, these notions were crude compared to modern notions of physical dimension; the Greeks belied one could not divide a volume by a length, for instance, [3].

The geometrical notion of dimension was extended by Fourier for the first time to include physical dimensions [4], “It must now be remarked that every undetermined magnitude or constant has one *dimension* proper to itself, and that the terms of one and the same equation could not be compared, if they had not the same *exponent of dimension*”

Dimensional analysis grew out of these ideas. It is applied, for instance, to check the correctness of equations in the sense that all terms should have the same dimensions. Moreover, a change in the systems of units employed to measure the different quantities involved in a given equation does not change original relationship given the equation. For the mathematical form of Newton’s second Law: $Force = Mass \times Acceleration$ must always hold regardless of the units of measurement used for the physical quantities of $Force$, $Mass$ and $Acceleration$. The mathematical law therefore constrains the dimensions of at least one of these quantities such that dimensional consistency is maintained.

Dimensional analysis is also utilized in the derivations of relations between physical magnitudes applying the principle of homogeneity. For instance, it is possible to derive (except for a dimensionless constant) the dependence of the frequency of oscillation of a pendulum near the earth’s surface on the pendulum’s length and on the earth’s gravitational field by considering only the dimensions or units of these physical terms.

## 2.2 The Buckingham Pi-Theorem

Dimensions place inviolate constraints on the possible combinations of parameters within an equation in order for validity to be maintained. The principle of dimensional homogeneity guarantees that, in every possible and correct physical equation, the dimensions on the left hand side of the equal sign are identical to those on the right hand side. Due to this property of all possibly correct physical functions $f(x_1,\ldots,x_n)=0$, the Pi-Theorem of Buckingham [1] can be derived. This theorem is stated below:

**Theorem 1-1 (Pi-Theorem):** *From the existence of a complete and dimensionally homogenous function $f$ of $n$ physical quantities $x_i \in \Re$ follows the existence of a dimensionless function $F$ of $m \le n$ dimensionless quantities $\pi_j \in \Re$*

$$f(x_1,\ldots,x_n)=0 \tag{2.1}$$

$$F(\pi_1,\ldots,\pi_m)=0 \tag{2.2}$$

*where $m=n-r$ is the number of dimensional quantities reduced by the rank $r$ of the dimensional matrix formed by the $n$ dimensional quantities. The dimensionless quantities (also dimensionless products or dimensionless groups) have the form*

$$\pi_j = x_{j+r}\prod^{r} x_i^{-\alpha(j+r)i} \qquad j=1,\ldots,m \tag{2.3}$$

*for $j=1,\ldots,m \in N^+$ and with the $\alpha_{ji} \in \Re$* as constants.

**Proof:** The proof of the Pi-Theorem is given in [1, 5]. ■

## 2.3 Examples

The expression of dynamic systems in dimensionless form changes neither the nature of the equation nor the dynamic principles upon which they are derived. The physics of a system dictate a mathematical relationship or rule that governs the different physical quantities involved in the mathematical equation. This validity is not dependent on the chosen units of measurement. In other words, scaling of the variables in an equation changes only the units of measurement, therefore the equation remains true irrespective of such scaling.

For example recall the Newton's second law: $F = M \times a$ with $F \equiv Force$, $M \equiv Mass$ and $a \equiv Acceleration$. Furthermore, assume that $F$ is measured in $Newtons$, $M$ in $kilograms$ and $a$ in $\frac{meters}{(second)^2}$. If the equation is scaled by the physical quantity $M$, the new equation becomes $f = a$ where $f = \frac{F}{M}$. This new equation is valid and is another form of Newton's second law, and the only difference is that the quantity $f$ is measured in $\frac{Newtons}{kilogram}$, in the new equation representing the ratio of the units of force to the units of mass.

The dimensionless representation of systems is illustrated using the mass-spring-damper system below. The equation of motion of the mass spring damper system shown in Figure **2-1** is given by Eq. **2.4**, an equation of three parameters and one input. Without affecting its mathematical relevance and physical meaning, Eq. **2.4** can also be represented in many ways. For instance, dividing the equation by the mass of the block yields another equation with only two parameters and one input (Eq. **2.5**), yet with the same meaning and physics.

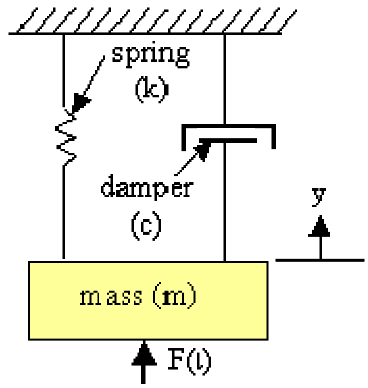



Figure **2-1**: Mass-spring-damper system

$$m\ddot{y} + c\dot{y} + ky = F \qquad \textbf{2.4}$$

$$\ddot{y} + \frac{c}{m}\dot{y} + \frac{k}{m}y = \frac{1}{m}F \qquad \textbf{2.5}$$

Finally, expressing each variable in a dimensionless form still maintains the meaning but results in a more compact mathematical expression regarding the number of variables involved. This is given by Eq. **2.6** and involves time scaling.

$$\bar{y}'' + \pi \cdot \bar{y}' + \bar{y} = \bar{F} \qquad \textbf{2.6}$$

where, $\tau = t\sqrt{\frac{k}{m}}$, $\pi = \frac{c}{\sqrt{mk}}$, $\bar{y} = y\frac{k}{mg}$, $\bar{F} = F\frac{1}{mg}$ and the new time derivative is $(') \equiv \frac{d}{d\tau} = \sqrt{\frac{m}{k}}\frac{d}{dt}$.

Comparing Eq. **2.4** and Eq. **2.6**, there are three parameters $\{m, c, k\}$ and two variables $\{y, F\}$ in the previous equation and one parameter $\{\pi\}$ and two variables

$\{\bar{y}, \bar{F}\}$ in the later. The dimensionless form of the system compactly represents groups of systems that share similar dynamic properties so that, thereafter, these systems can be analyzed as a group.

## 2.4 References


[1] E. Buckingham, "On Physically Similar Systems; Illustration of the Use of Dimensional Equations," in *Physical Review*, vol. 4, *2nd series*, 1914, pp. 345-376.

[2] E. Buckingham, "The Principle of Similitude," in *Nature*, vol. 96, 1915, pp. 396-397.

[3] R. d. A. Martins, "The Origin of Dimensional Analysis," *Journal of the Franklin Institute*, vol. 311, pp. 331-337, 1981.

[4] J. B. J. Fourier, "Analytical theory of heat," in *Great Books of the Western World*, vol. 45. Chicago: Encyclopaedia Britanica, 1952, pp. 161-251.

[5] P. W. Bridgman, *Dimensional Analysis*, 3rd printing, revised edition ed. New Haven: Yale University Press, 1943.

# Chapter 3

# Dimensional Transformation and Control Problems

## 3.1 Introduction

This chapter discusses the method of dimensional transformation and how it can be used in control problems. A step-by-step formulation of the transformation is presented. This is used as a tool for later chapters that present the use of the dimensional transformation as a means to optimize the system representation for gain-scheduling and robust control designs.

Dimensional analysis is widely used in a range of engineering disciplines. Not surprisingly, some authors have formulated gain-scheduled and robust control laws in dimensionless representations. For example, Corriga, et. al [1] has used dimensionless representation for implicit gain-scheduling via time scaling. Tavakoli [2] and Astrom [3] have used dimensional analysis to normalize important parameters in the tuning of PID controllers. Brennan and Alleyne [4, 5] have used dimensional analysis for robust vehicle control.

## 3.2 The gantry problem: a working example

A gantry model is chosen in this work to serve as an example of the dimensional transformation and gain-scheduling topics presented. Gantries are implemented in many applications ranging from industrial overhead cranes to harbor load-handling systems. Their dynamics are simple enough to be easily derived from first principles, yet they are complex enough parametrically that their operation requires scheduling with respect to operating conditions. A standard gantry loading cycle involves lifting a load, moving the load to a different position (perhaps while lifting and/or lowering), lowering the load in a new position, then returning back to original position to start a new cycle. This process generally requires two or more of the parameters of the system to vary in a given cycle, namely the varying payload on the gantry, $m_p$ , the varying swing-length of the pendulum, $L$ , and sometimes a varying mass of the trolley, $m_t$ due to uptake of a heavy cable. Recent work on crane control by Corriga, et. al [1] focusing on implicit gain-scheduling via time scaling can be considered special case of reduction of gain-scheduling parameters for this particular example.

The gantry system is modeled as a pendulum attached to a moving trolley as shown in Figure **3-1**. The equation of motion of the gantry system, as derived in Franklin, et. al. [6] and further assuming $I$ to be very small compared to $m_p L^2$, is given by Eq. **3.1**.

$$
\begin{aligned}
&(m_t + m_p)\ddot{x} + b\dot{x} + m_p L\ddot{\theta}\cos(\theta) - m_p L\dot{\theta}^2\sin(\theta) = u \\
&L\ddot{\theta} + g\sin(\theta) = -\ddot{x}\cos(\theta)
\end{aligned}
\qquad \textbf{3.1}
$$

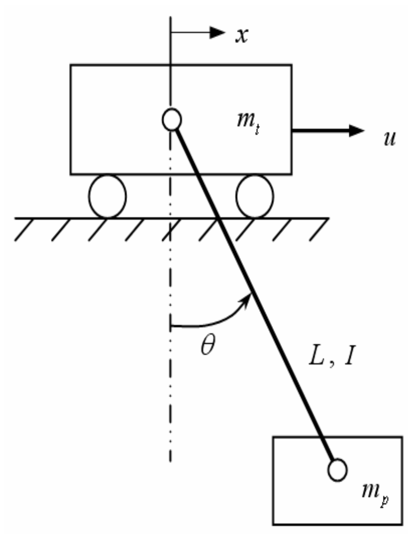



Figure **3-1**: Schematic of the gantry system

For small motions of $\theta$ about the equilibrium position $\theta = 0$, it can further be assumed that $\sin(\theta) \approx \theta$, $\cos(\theta) \approx 1$ and $\dot{\theta}^2 \approx 0$. Using these assumptions, a linearized form of Eq. **3.1** is given by Eq. **3.2**.

$$\begin{aligned} &(m_t + m_p)\ddot{x} + m_p L\ddot{\theta} + b\dot{x} = u \\ &\ddot{x} + L\ddot{\theta} + g\theta = 0 \end{aligned} \qquad \textbf{3.2}$$

In state-space form, defining $x_1 = x$; $x_2 = \theta$; $x_3 = \dot{x}$ and $x_4 = \dot{\theta}$, Eq. **3.2** can be written as, $\dot{\mathbf{x}} = A\mathbf{x} + Bu$, where $A$ and $B$ are the system matrices given by Eq. **3.3**.

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \dfrac{m_p g}{m_t} & -\dfrac{b}{m_t} & 0 \\ 0 & -\dfrac{g}{L}\left(1 + \dfrac{m_p}{m_t}\right) & \dfrac{b}{m_t L} & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ 0 \\ \dfrac{1}{m_t} \\ -\dfrac{1}{m_t L} \end{bmatrix} \qquad \textbf{3.3}$$

In practice, either the nonlinear equations of Eq. **3.1** or the linear equations of Eq. **3.2** could be used for gain-scheduled controller design using the method of dimensionless representation, but for simplicity we utilize the linearized model in later sections.

### 3.3 The dimensional transformation procedure

In this section, the process of dimensional transformation is generalized and presented, with the gantry example used to illustrate the mechanics of the transformation process. Before, generalizing the method consider the example of expressing the input force ($u$) of the gantry system in a dimensionless form ($\overline{u}$). Suppose that the parameters $g, m_p$ and $L$ are chosen as the scaling parameters. Hence, the general expression of the dimensionless force in terms of the dimensional force and the scaling parameters is given by Eq. **3.4**.

$$\overline{u} = u \cdot g^a \cdot m_p^b \cdot L^c \qquad \textbf{3.4}$$

where, $a, b$ and $c$ are unknowns. These unknowns are to be determined from the consistency of the units of measurement. Assume that the basis units used to span all the variables in Eq. **3.4** are *length*, *mass* and *time* represented in short as $l, m$ and $t$, respectively. The unit consistency equation is given by Eq. **3.5**.

$$\begin{aligned}
&[\overline{u}] = [u] \cdot [g]^a \cdot [m_p]^b \cdot [L]^c \\
&\Rightarrow l^0 \cdot m^0 \cdot t^0 = \left(l^1 \cdot m^1 \cdot t^{-2}\right) \cdot \left(l^1 \cdot m^0 \cdot t^{-2}\right)^a \cdot \left(l^0 \cdot m^1 \cdot t^0\right)^b \cdot \left(l^1 \cdot m^0 \cdot t^0\right)^c \\
&\Rightarrow l^0 \cdot m^0 \cdot t^0 = l^{1+a+c} \cdot m^{b+1} \cdot t^{-2-2a}
\end{aligned} \qquad \textbf{3.5}$$

Equating the exponents of the last equation in Eq. **3.5** yields Eq. **3.6**.

$$\begin{aligned} a + c + 1 &= 0 \\ b + 1 &= 0 \\ -2a - 2 &= 0 \end{aligned} \qquad \mathbf{3.6}$$

and the solution is, $a = -1,\ b = -1$ and $c = 0$. With this solution, the dimensionless force ($\bar{u}$) can be expressed as: Eq. **3.7**.

$$\bar{u} = u \cdot g^{-1} \cdot m_p^{-1} \cdot L^0 = \frac{u}{m_p g} \qquad \mathbf{3.7}$$

To generalize dimensional analysis to system theory, consider a functional relation of an actual physical system as $f(x_1,...x_{N_x}, p_1,..., p_{N_p}) = 0$, where the $x_i$'s and $p_i$'s are signals and parameters as defined in Table **3-1**.

Table **3-1**: The dimensional transformation process

| | signals: $x_1$ $x_2$ $\cdots$ $x_{N_x}$ ; parameters: $p_1$ $p_2$ $\cdots$ | parameters: $\cdots$ $p_{N_p-1}$ $p_{N_p}$ |
|---|---|---|
| $e_1$, $e_2$, $\vdots$, $e_{N_e}$ | $B_D$ | $A_D$ |
| $\pi_1$, $\pi_2$, $\vdots$, $\pi_{N_\pi}$ | $I$ | $C_S$ |

The Buckingham-Pi theorem proves that this equation can also be represented in an information-equivalent form as $\bar{f}(\pi_1, \pi_2, ..., \pi_{N_\pi}) = 0$, where the $\pi_i$'s are formed by grouping of the $x_i$'s and $p_i$'s and are dimensionless. The Pi-Theorem states that

$N_\pi \leq N$, where $N_\pi$ is the number of variables in the dimensionless formulation (usually called $\pi$ variables), and $N$ is the number of variables in the original formulation. In other words, there are usually less variables required to express the equations of a physical law, for instance equations of motion, than when dimensioned representations are used. To formalize the process of a dimensional transformation, consider any functional description of dynamic behavior (a plant or controller) dependent on $N$ variables. If the description is a dynamic one, the variables will generally span unit dimensions of length, mass, and time. Hereafter we assume that the units of each variable $v$ can be written as a vector that is extracted via a dimensional extraction operator, $d_{v,e} = D(e,v)$. To uniquely define this vector, one must specify both the unit space as well as the parameter. For instance, the gravitational constant, $g = 9.81\ \mathrm{m/s^2}$, has dimensional units that can be represented in one unit system, $e = [length \quad mass \quad time]^T$, as a column vector, $d_{v,e} = D(e,g) = [1 \quad 0 \quad -2]^T$, or in another unit system of $e = [mass \quad force]^T$, as $d_{v,e} = D(e,g) = [-1 \quad 1]^T$.

The dimensional unit system is known to be an arbitrary factor in representing a system [7], therefore we seek to rescale the system by selecting unit systems that give specific advantage to the gain-scheduling or robust control problem. Some unit systems are clearly advantageous for controller design purposes, particularly ones producing dimensionless representations (see Brennan [5]). The transformation from/to a dimensioned to/from a dimensionless system is fairly straightforward and follows a procedure similar to a simple basis transformation (see Szirtes [8]). This process is a modern formalizations of the unit normalization procedure first described in the

Buckingham-Pi Theorem [8] in his seminal paper nearly a century ago. The details of the derivation can be found in [5].

The Matrix $A_D$ in Table **3-1** is formed by the dimensional unit vectors of the variables (parameters) that are chosen as scaling parameters, also traditionally called repeating parameters. $A_D$ is a square matrix with its rows (and column) size equal to the row size of $e$ and must be a full rank. The matrix $B_D$ is formed by the dimensional unit vector of the remaining system variables that are to be scaled, also traditionally called non-repeating variables.

First, one notes the vector representing the units of each parameter or signal, hereafter called a variable, using the $d_{v,e} = D(e, v)$ operation. One then writes the resulting vectors as column of the matrices $B_D$ or $A_D$ as shown in Table **3-1** [5]. The choice of variables to select in these matrices, and their ordering, is discussed shortly. For clarity, the vectors are labeled overhead with the variable whose dimensions they represent. In the case of Table **3-1**, the general unit system is selected as, $e = \begin{bmatrix} e_1 & e_2 & \dots & e_{N_e} \end{bmatrix}^T$. For example, in Table **3-1**, if $p_1 = g$ (the gravitational constant), then for a unit system $e = \begin{bmatrix} e_1 & e_2 & e_3 \end{bmatrix}^T = \begin{bmatrix} length & mass & time \end{bmatrix}^T$, the $N_x+1$ column of $B_D$ would be $d_i = [1 \quad 0 \quad -2]^T$. If $p_{N_p} = g$ instead, then the last column of $A_D$ would be $d_i = [1 \quad 0 \quad -2]^T$, and so on. This matrix representation $[B_D \; A_D]$ will always have $N_e$ rows and $N_x + N_p$ columns because each variable has one dimensional unit vector. Hence the size of this vector is equal to $N_e$. For clarity, the variable labels are shown at top of the matrices and unit dimensions are shown at left. Also note that the $x_i$'s represents the signals (time, state variables, input signals, etc.) whereas $p_i$'s are parameters. In this process, variables forming columns in matrix $A_D$ are considered the scaling variables whereas those forming columns in matrix $B_D$ are called scaled variables.

The challenge of dimensional scaling is to choose variables to participate in the scaling matrix $A_D$ such that the transformed system is most amenable to gain scheduling and/or robust control. To choose scaling variables, one rearranges the columns of the matrices $A_D$ and $B_D$, selecting $N_e$ variable-columns for $A_D$ such that the corresponding columns are linearly independent, i.e. they can together form the full rank matrix $A_D$ (rank = $N_e$). This rearrangement is always feasible if the unit system is not redundant (see Szirtes [8]). By rearranging the matrices one is simply changing the scaling parameters also called repeating parameters. When choosing scaling parameters it is important to keep in mind the following three points:

1. The resulting scaling matrix $A_D$ should be non-singular.
2. For dynamic systems the use of signals as scaling variables should be avoided especially in linear systems because the problem becomes mathematically harder to solve than the original form. For example consider the mass-spring-damper model given in Eq. **3.8**.

$$m\ddot{y} + c\dot{y} + ky = F \qquad \textbf{3.8}$$

Dividing by the dependent variable $y$, results in a nonlinear form as in Eq. **3.9** and obviously harder to solve.

$$m\frac{\ddot{y}}{y}+c\frac{\dot{y}}{y}+k=\frac{1}{y}F \qquad \textbf{3.9}$$

From a dimensional analysis viewpoint, the choice of which variables to place in $A_D$ is a user-defined choice with the *only* restriction being the first rule described above. From a control theory standpoint, in addition to the first rule above, the user should also generally avoid using signals as scaling for the reason given by the 2nd rule above.

For convenience, we hereafter assume that the matrices $A_D$ and $B_D$ are arranged such that the $N_e$ columns of $A_D$ occur on the $N_e$ right-most columns. This allows the partitioning of the matrix in the form given in Table **3-1** where, $e_i$'s are the dimensional basis vectors that span the $x_i$'s and $p_i$'s. To represent the variables of $B_D$ in the new dimensional basis given by the column vectors associated with $A_D$, one calculates the matrix given by Eq. **3.10**. The details of derivation for the general case is given in [5]. In general, Eq. **3.10** is the matrix form of solving the system of linear equations of the form given in Eq. **3.5** and applied to all the variables in the system model of interest.

$$C_S=\left(-A_D^{-1}.B_D\right)^T \qquad \textbf{3.10}$$

The lower left partition of Table **3-1** is always unity to generate dimensionless parameters. This process of dimensional transformation is demonstrated on the gantry problem, and results are presented in Table **3-2**. The bottom-left rows of the partitioned matrix in Table **3-2** indicate the dimensionless variables (signals and parameters) of the system, with the number of dimensionless variables, $N_\pi$ , given by $N_\pi = N_x + N_p - N_e$, and is equal to six in the gantry example (by inspection).

Table **3-2**: Example of the dimensional transformation process

| | $b$ | $m_t$ | $x$ | $\theta$ | $u$ | $t$ | $g$ | $m_p$ | $L$ |
|---|---|---|---|---|---|---|---|---|---|
| $m$ | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| $kg$ | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| $s$ | $-1$ | 0 | 0 | 0 | $-2$ | 1 | $-2$ | 0 | 0 |
| $\pi_1$ | 1 | 0 | 0 | 0 | 0 | 0 | $-1/2$ | $-1$ | $1/2$ |
| $\pi_2$ | 0 | 1 | 0 | 0 | 0 | 0 | 0 | $-1$ | 0 |
| $\pi_3$ | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | $-1$ |
| $\pi_4$ | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| $\pi_5$ | 0 | 0 | 0 | 0 | 1 | 0 | $-1$ | $-1$ | 0 |
| $\pi_6$ | 0 | 0 | 0 | 0 | 0 | 1 | $1/2$ | 0 | $-1/2$ |

Each bottom row indicates how new dimensionless variables, hereafter called $\pi$-variables, should be created from powers of each of the column variables of $A_D$. As a specific example, the first row gives the first $\pi$-variable: $\pi_1 = b^1 \cdot g^{-1/2} \cdot m_p^{-1} \cdot L^{1/2}$. The entire $\pi$-variables of the gantry (scaled signals and parameters) are shown in Eq. **3.11**.

$$\begin{gathered} \pi_1 = \frac{b}{m_p}\sqrt{L/g}, \qquad \pi_2 = \frac{m_t}{m_p}, \qquad \pi_3 = \frac{x}{L} = \bar{x}, \\ \pi_4 = \theta.1 = \bar{\theta}, \quad \pi_5 = \frac{u}{m_p g} = \bar{u}, \quad \pi_6 = t\sqrt{g/L} = \tau \end{gathered} \qquad \mathbf{3.11}$$

In the gantry example, the choice of scaling parameters $m_p$, $L$ and $g$ to form a new dimensional basis imposes a new time-scaling of $\sqrt{g/L}$, a mass scaling of $1/m_p$, and a length scaling of $1/L$. Note that the first two $\pi$-variables are associated with time varying parameters (parameters requiring gain-scheduling), and the last four are associated with signals. The original system had three varying parameters, namely the varying payload on the gantry, $m_p$, the varying swing-length of the pendulum, $L$, and the varying mass of the trolley, $m_t$, while clearly the dimensionless representation has only two varying parameters $\pi_1$ and $\pi_2$.

The gantry example illustrates that a reduction in the number of parameters via dimensional transformation, but it does not specifically address how to select variables to form $A_D$ to obtain best reduction in variable dependence. This topic is discussed in detail in chapter 4.

## 3.4 The dimensionless system representation

The dimensional transformation method, in general, transforms a system description with a given dimensional basis units representation to another system description with another dimensional basis units representation as chosen by the user. In this particular case, the second dimensional basis units representation was chosen to be unity (dimensionless), and hence the transformed system will be a dimensionless form of the original system description. In other words, all signals (states) and parameters in the dimensionless representation are normalized through variable grouping and are dimensionless. The choice of dimensionless representation is due to the advantage that it gives the minimum number of variable dependence in a given system representation. Based on the dimensional transformation results given in Eq. **3.11**, the gantry system previously represented in dimensional description, Eq. **3.1**, is transformed into the dimensionless description, given by Eq. **3.12**.

$$\begin{aligned} &(\pi_2+1)\bar{x}''+\pi_1\bar{x}'+\bar{\theta}''\cos(\bar{\theta})-\sin(\bar{\theta})=\bar{u} \\ &\bar{\theta}''+\sin(\bar{\theta})=-\bar{x}''\cos(\bar{\theta}) \end{aligned} \qquad \textbf{3.12}$$

where a time normalization is applied to the derivative operator, i.e. $(') \equiv \dfrac{d}{d\tau} = \sqrt{L/g}\,\dfrac{d}{dt}$ .

The linearized form of Eq. **3.12** in state-space form is given by, $\bar{\mathbf{x}}' = \bar{A}\bar{\mathbf{x}} + \bar{B}\bar{u}$ , where the states are defined as $\bar{x}_1 = \bar{x}$; $\bar{x}_2 = \bar{\theta}$; $\bar{x}_3 = \bar{x}'$ and $\bar{x}_4 = \bar{\theta}'$, and the dimensionless matrices $\bar{A}$ and $\bar{B}$ given by Eq. **3.13**,

$$\bar{A} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1/\pi_2 & -\pi_1/\pi_2 & 0 \\ 0 & -(1+1/\pi_2) & \pi_1/\pi_2 & 0 \end{bmatrix} \quad \bar{B} = \begin{bmatrix} 0 \\ 0 \\ 1/\pi_2 \\ -1/\pi_2 \end{bmatrix} \qquad \textbf{3.13}$$

## 3.5 Equivalence of the dimensional and dimensionless control designs

It may appear questionable that the dynamics and controller design for a dimensioned representation is equivalently performed within a dimensionless framework. To illustrate that the dimensional transformation actually has no effect on the controller design, a full state-feedback controller is designed for the gantry example using a simple pole placement method [6] in both the dimensional ( $K$ ) and dimensionless ( $\bar{K}$ ) domain using system representations from Eq. **3.3** and Eq. **3.13**.

The physical parameters used in each design are $m_p = 0.196kg$ , $m_t = 1.21kg$ , $L = 0.311m$ , $b = 2.5kg/s$. A diagram of the feedback control structure for the two methods is represented in Figure **3-2**. The desired pole locations were chosen for this instance to be: $p = [-2.5 \pm i, \quad -1.5 \pm 2i]$ (the dimensionless poles are $\overline{p} = p.\sqrt{L/g}$ ), with resulting gains $K$ and $\overline{K}$ given in Eq. **3.14**

$$\begin{aligned} K &= [1.7137 \quad 3.7978 \quad -0.4955 \quad -2.3871] \\ \overline{K} &= [0.2733 \quad 1.9474 \quad -0.4470 \quad -6.9234] \end{aligned} \qquad \textbf{3.14}$$

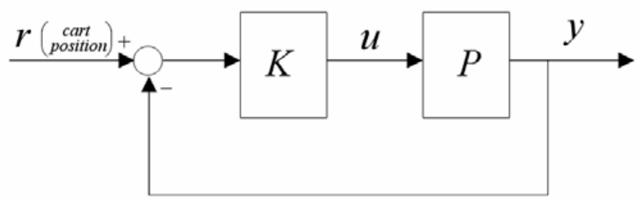



Figure **3-2**: The feedback control structure

While the gains are obviously different because of the different plant representations, the equivalence of controllers can be derived. Consider the state feedback controller in Eq. **3.15**.

$$u = K\mathbf{x} = k_1 x_1 + k_2 x_2 + k_3 x_3 + k_4 x_4 \qquad \textbf{3.15}$$

Substituting the dimensionless form of the input and states,

$$\begin{aligned} m_p g\overline{u} &= k_1 L\overline{x}_1 + k_2 \overline{x}_2 + k_3 \sqrt{\frac{g}{L}} L\overline{x}_3 + k_4 \sqrt{\frac{g}{L}} \overline{x}_4 \\ \Rightarrow \overline{u} &= \frac{k_1 L}{m_p g}\overline{x}_1 + \frac{k_2}{m_p g}\overline{x}_2 + \frac{k_3}{m_p \sqrt{g/L}}\overline{x}_3 + \frac{k_4}{m_p \sqrt{gL}}\overline{x}_4 = \overline{K}\overline{\mathbf{x}} \end{aligned}$$

where, the dimensionless gain vector is given by Eq. **3.16**.

$$\bar{K} = \begin{bmatrix} \frac{k_1 L}{m_p g} & \frac{k_2}{m_p g} & \frac{k_3}{m_p \sqrt{g/L}} & \frac{k_4}{m_p \sqrt{gL}} \end{bmatrix} \qquad \textbf{3.16}$$

The equivalence in the control design in the two representations: the dimensional and dimensionless domains can also be seen by comparison of tracking performance from both closed-loop systems, as shown by Figure **3-3**. The simulation is performed using SIMULINK.

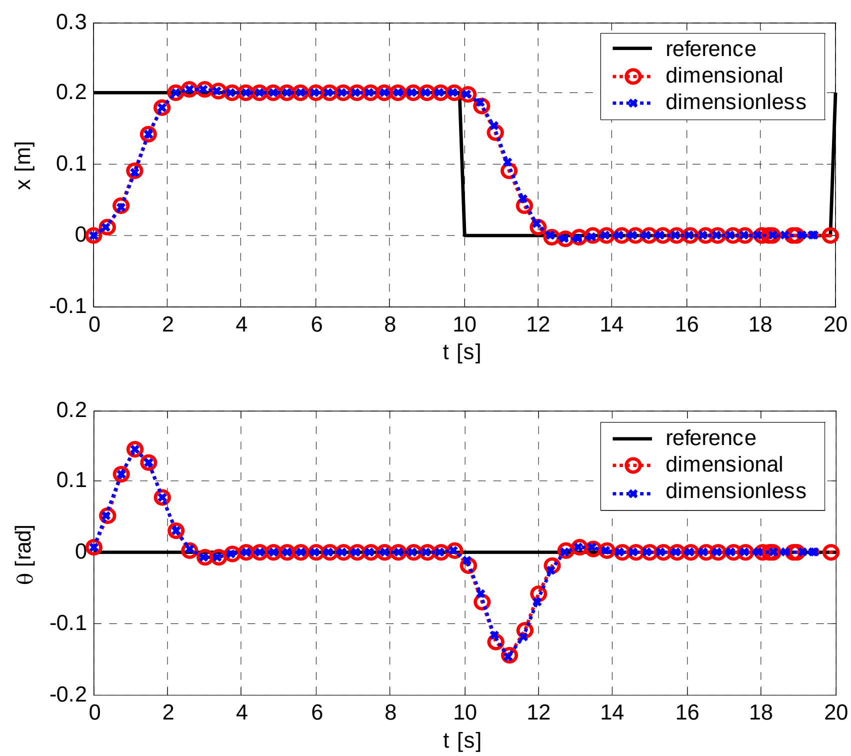



Figure **3-3**: Tracking response to a square wave input, *Top*: cart position, *Bottom*: angular position of the pendulum.

The time scales for the dimensional and dimensionless representations are different. Because of this and the fact that MATLAB has only single clock simulation the two systems cannot be simulated in parallel. Therefore, the simulation results shown in

Figure **3-3** are obtained by running the simulation of the dimensionless systems at a scaled frequency consistent to the time scaling. To drive this relationship, consider frequency and period of the dimensionless systems to be $\{\overline{\omega}, \overline{T}\}$and that of the dimensional representation $\{\omega, T\}$. Then, Eq. **3.17** always holds regardless of the representation.

$$\overline{\omega} \cdot \overline{T} = \omega \cdot T = 1 \tag{3.17}$$

From the definition of time scaling, $\tau = t \cdot \sqrt{g/L}$, the period scaling is calculated as $\overline{T} = T \cdot \sqrt{g/L}$. Substituting this into Eq. **3.17** yields Eq. **3.18**.

$$\overline{\omega} = \frac{T}{\overline{T}} \omega = \frac{\omega}{\sqrt{g/L}} \tag{3.18}$$

The $\overline{y}$ vector obtained from the simulation of the dimensionless systems can now be plotted as a function of the real time $t$ to get the corresponding dimensional data for comparison with the dimensional system. Note that even with the frequency scaling, it is not possible to scope both plots on the same scale. Therefore, the both vectors are saved to workspace and plotted against the same real time vector.

More dimensional transformation methods especially suited for dynamic controllers are discussed in chapter 6.

### 3.6 References

[1] G. Corriga, A. Giua, and G. Usai, "An Implicit Gain-Scheduling Controller for Cranes," *IEEE Transactions on Control Sysytems Technology*, vol. 6, pp. 15-20, 1998.

[2] S. Tavakoli and M. Tavakoli, "Optimal Tuning of PID Controllers for First Order plus Time Delay Models Using Dimensional Analysis," The 4th IEEE International Conference on Control and Automation, Canada, 2003.

[3] K. J. Astrom, C. C. Hang, P. Persson, and W. K. Ho, "Towards Intelligent PID Control," *Automatica*, vol. 28, pp. 1-9, 1992.

[4] S. Brennan and A. Alleyne, "Generalized H-infinity Vehicle Control Utilizing Dimensional Analysis," American Control Conference, Denver, CO, 2003.

[5] S. N. Brennan, "On Size and Control: The Use of Dimensional Analysis in Controller Design," in *Mechanical Engineering*. Urbana-Champaign: University of Illinois, 2002, pp. 370.

[6] G. F. Franklin, J. D. Powell, and A. Emami-Naeini, *Feedback Control of Dynamic Systems*, 3 ed: Addison-Wesley Publishing Company, 1994.

[7] P. W. Bridgman, *Dimensional Analysis*: Yale University Press, 1931.

[8] T. Szirtes, *Applied Dimensional Analysis and Modeling*: McGraw-Hill, 1998.

# Chapter 4

# Gain-Scheduling and Parametric Space Minimization

## 4.1 Introduction

This chapter presents a technique that minimizes the number of gain scheduling parameters to potentially reduce the complexity and practical limitation of the gain scheduling control synthesis. This chapter also discusses the development of a mathematical formulation relating the reduction in gain scheduling parameters to the choice of scaling parameters. An optimization algorithm is also developed for searching for the optimum scaling matrix that results in the minimum possible scheduling parameters.

In systems that require gain scheduling due to changes in systems parameters, the gain-scheduling controller synthesis is performed for all varying parameters and along the entire path of variation of each parameter. As will be discussed later in this chapter, a large number of varying parameters can impose a huge burden on the gain-scheduling control synthesis. This in turn limits the practical application of the gain-scheduling technique only to scheduling with respect to fewer parameters [1].

Gain scheduling is one of the most popular controller design approaches for systems whose model dynamics vary widely over their range of operation. Examples include nonlinear system and linear parameter-varying (LPV) systems. Linear controllers that are designed based on a fixed plant dynamic model that usually assumed to be fully known, have several limitations. These limitations include 1) lack of robustness to disturbances and noise, 2) deteriorated performance or sometimes instability for small plant perturbations, 3) lack of adaptability to changing operating condition such as changing system parameters and/or environment, etc. Therefore, other advanced controller techniques are sought for systems that do not have a fixed operating point or systems whose performance requirements cannot be satisfied with such controller designs.

In the instance where the actual operating point of the plant lies within a small and known-bound set of plant perturbations, a single controller can be designed based on a selected model in the set, called the nominal model, to robustly stabilize and satisfy performance requirements for all the perturbations in the bounded set (also called the uncertainty set). Such control design approach is known as the robust control design.

However, not all systems exhibit only small perturbations. For instance, when a system to be controlled is nonlinear and its dynamics changes significantly during operation and the operating conditions (such as system parameters) changes considerably in its course of operation, it may not be possible to design a linear or robust controller for this wide range of operation. In such cases, the gain scheduling control design approach is often used. This method has been widely applied in the aerospace and process control applications [2].

## 4.2 Literature Review

Gain scheduling is a wide area of research with extensive literature focusing on several methods. Selected and related publications are discussed to give enough background and motivation for the current work.

There are several methods to implement a gain-scheduling algorithm, including switching control gains as a function of operating conditions (see for e.g. Shimomura [3]) or interpolating between different linear control laws (see for e.g. Rugh [4]). With new mathematical software packages, symbolic formulations of dynamic equations are easier than ever to derive and manipulate, so control systems – both linear and nonlinear – are increasingly gain-scheduled directly through the use of parametric controller synthesis. This results in either linear parameter varying (LPV) controllers or nonlinear control laws [1, 5-7].

Despite these advances in control synthesis, there are many limitations, both theoretical and practical, as to the number of gain scheduling parameters that can be simultaneously changed in a control law [7, 8]. In most cases the problem complexity increases exponentially with the number of scheduling parameters [1, 7].

Dimensional analysis is widely used in a range of engineering disciplines and, not surprisingly, some authors have formulated gain-scheduled control laws in dimensionless representations. For example, Corriga, et. al [9] has used dimensionless representation for implicit gain-scheduling via time scaling. Tavakoli [10] and Astrom [11] have used dimensional analysis to normalize important parameters in the tuning of PID controllers. The motivation for such presentation of dimensionless controllers is to present the most compact and general solution. This work extends such prior work significantly in that the dimensional transformation process itself is considered as a means to optimize the system representation for gain-scheduling controller design.

### 4.3 Dimensional Transformation and the Gain Scheduling Problem

Figure **4-1** shows the generalized representation of a *v*-parameter scheduling of a controller and system. In this representation, the system and controller dynamics are dependent on $V$ gain scheduling parameters, $p_1, p_2, ..., p_V$. Additionally, the plant is often dependent on a larger set of $N$ parameters that include both the varying, gain-scheduled parameters as well as $R = N - V$ constant parameters , i.e. the plant can be parameterized as $P\left(p_1, p_2, ..., p_V, p_{V+1}, ..., p_N\right)$. The intent of gain-scheduling is therefore simultaneously change the control law $K\left(\cdot\right)$ as a function of the same time-varying parameters affecting the plant, namely $p_1, p_2, ..., p_V$ , as shown in Figure **4-1**.

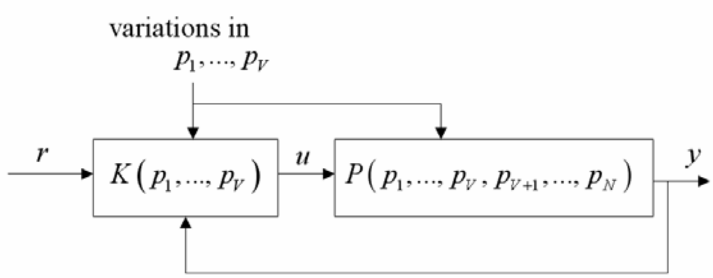



Figure **4-1**: Controller and plant both scheduled to varying parameters

To formulate the control laws, a gain-scheduled controller of form $K\left(p_1,\ldots,p_V\right)$ is obtained either by maintaining a symbolic dynamic representation throughout a controller-synthesis algorithm or by developing user-defined heuristic functions. When user-defined functions are used, they are often obtained from interpolating or extrapolating from point-to-point controller designs obtained at fixed values of the scheduling variables. User-defined heuristic functions are therefore useful in that they do not require manipulation of symbolic equations in a controller synthesis. In some cases, they do not even require a symbolic plant representation. This allows gain-scheduling to remain a useful control technique for systems whose dynamic representations are too complex, or for controller synthesis procedures that are too complex, to reasonably permit closed-form algebraic solutions.

When point-wise control synthesis techniques are used to obtain gain-scheduled controllers, the task of controller design and/or controller evaluation grows exponentially more cumbersome with increasing number of scheduling parameters. If the number of scheduling parameters is $V$ and the number of levels of each variable to be considered is $L$, then the number of different controller designs and/or analyses, $N_K(V,L)$, is given by Eq. **4.1**

$$N_K(V,L) = L^V \qquad \textbf{4.1}$$

For example, if a system is analyzed at two fixed levels of each scheduling variable ($L=2$), then a three parameter gain-scheduling controller ($V=3$) will require eight designs and/or analyses. Choosing a more realistic assumption where a nonlinear or LPV system requires analysis at more than just two levels of each variable – ten levels for instance – then the complexity grows exceedingly quickly: a three variable problem requires 1000 separate control design syntheses, a four variable problem requires 10,000 designs, etc. This exponential growth in controller complexity clearly limits the number of scheduling variables that can be easily considered [1, 8, 12] in design or analysis of common gain scheduling algorithms.

Here we present a technique to reduce the number of gain-scheduled variables by attempting to reformulate gain-scheduling problems in a lower number of scheduling variables using dimensional transformation as shown in Figure **4-2**.

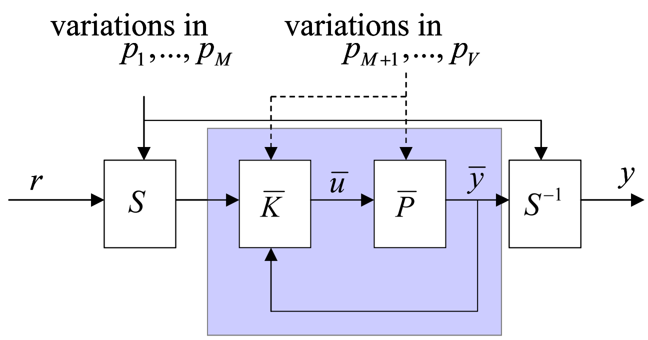



Figure **4-2**: Scheduling controller and scaling parameters

## 4.4 Guaranteed reduction of parametric space for gain scheduling

The process of forming a dimensionless system representation is now scrutinized to obtain a generalized method of best selecting scaling parameters for a gain-scheduling controller design.  When both varying and non-varying groups of parameters are present in the gain-scheduling control problem, as is the case in almost all practical problems, the best choice of the $A_D$ and $B_D$ matrices is not straightforward to determine. One wish to choose repeating parameters to best map the higher space of gain scheduling parameters to a reduced, lower-dimensionality space. Further, one would like to obtain explicit bounds on the possible simplification of the problem. A careful selection of parameters involved in these matrices results in the minimum possible number of gain-scheduling parameters. But an arbitrary selection of parameters may make the gain scheduling problem worse.

More exactly, this chapter attempts to answer the following questions: When is a dimensionless representation beneficial in terms of gain-scheduling? Can the reduction in parameters be quantified by inspection? What criteria govern which parameters should be chosen as scaling parameters?

To answer the above questions, consider the dimensional transformation problem given in Table **4-1** representative of unit-scaling applied to a general gain-scheduling system. Assume that the control problem whose parameters are represented in Table **4-1** contains signals ($x_1 ... x_{Nx}$), varying parameters ($p_1 ... p_{Np}$) as well as non-varying parameters ($q_1 ... q_{Nq}$). The $A_D$ and $B_D$ matrices are now partitioned with respect to signals, non-varying parameters and varying parameters. The solution to the dimensional scaling follows from the previous chapter producing a scaling matrix $C_S$. This solution can now be partitioned into three row sub-matrices as defined in Eq. **4.2**.

Table **4-1**: Details of the dimensional transformation process

| | *nonvarrying parameters* | *varryingparameters* | *signals* | *nonvarrying parameters* | *varryingparameters* |
|---|---|---|---|---|---|
| | $q_1 \cdots q_{Nq1}$ | $p_1 \cdots p_{Np1}$ | $x_1 \cdots x_{Nx}$ | $q_{Nq1+1} \cdots q_{Nq}$ | $p_{Np1+1} \cdots p_{Np}$ |
| $e_1 \vdots e_{Ne}$ | $B_D^{(q1)}$ | $B_D^{(p1)}$ | $B_D^{(x)}$ | $A_D^{(q2)}$ | $A_D^{(p2)}$ |
| $\pi_1 \vdots \pi_{Nq1}$ | $I_{Nq1}$ | $0$ | $0$ | $C_S^{(q1)(q2)}$ | $C_S^{(q1)(p2)}$ |
| $\pi_{Nq1+1} \vdots \pi_{Nq1+Np1}$ | $0$ | $I_{Np1}$ | $0$ | $C_S^{(p1)}$ | |
| $\pi_{Nq1+Np1+1} \vdots \pi_{N\pi}$ | $0$ | $0$ | $I_{Nx}$ | $C_S^{(x)}$ | |

$$C_S = \left(-A_D^{-1} B_D\right)^T = -\begin{bmatrix} (B_D^{(q1)})^T A_D^{-T} \\ (B_D^{(p1)})^T A_D^{-T} \\ (B_D^{(x)})^T A_D^{-T} \end{bmatrix} = \begin{bmatrix} C_S^{(q1)(q2)} \;\vdots\; C_S^{(q1)(p2)} \\ C_S^{(p1)} \\ C_S^{(x)} \end{bmatrix} \qquad \textbf{4.2}$$

There are four matrices considered in $C_S$: $C_S^{(x)}$, $C_S^{(p1)}$, $C_S^{(q1)(q2)}$ and $C_S^{(q1)(p2)}$. Each has different meaning regarding the relationship between varying and non-varying variables in the system representation. By understanding each matrix, we can quantify whether dimensional scaling is beneficial or not beneficial in reducing parametric dependence to the benefit of gain scheduling.

The last row sub-matrix of the scaling matrix $C_S$ (i.e., $C_S^{(x)}$) scales only varying signals to obtain the dimensionless groups, $\pi_{Nq1+Np1+1} \ldots \pi_{N\pi}$. Since the signals $x_1 \ldots x_{Nx}$ are varying prior to transformation, scaling of these signals by either varying or non-varying parameters does not affect whether they vary after transformation. Thus, these do not affect the total number of varying parameters in the re-parameterized system.

Similarly, the second row sub-matrix ($C_S^{(p1)}$) is used to scale the varying parameters $p_1 \ldots p_{Np1}$. Regardless of whether these are scaled using varying or non-varying parameters, they will produce varying dimensionless groups $\pi_{Nq1+1} \ldots \pi_{Nq1+Np1}$. Hence, this matrix has no effect on the reduction of varying parameters.

The first row matrix is partitioned into two-column sub-matrix $C_S^{(q1)(q2)}$ and $C_S^{(q1)(p2)}$, as indicated in equation Eq. **4.2**. The first column sub-matrix of this ($C_S^{(q1)(q2)}$) corresponds to scaling non-varying parameters with other non-varying parameters. This will always result in non-varying parameters in the re-scaled system.

Finally, the $C_S^{(q1)(p2)}$ is the sub-matrix that represents scaling of non-varying parameters with varying parameters. This is the only scaling operation that will create new varying parameters in the re-scaled system from what were constant parameters in the original system representation. This is exactly opposite to the intent of using dimensional transformation, where the goal is to reduce the number of varying parameters, not create new ones. The number of these newly created varying parameters is equal to the row rank of the matrix $C_S^{(q1)(p2)}$ because it corresponds to the number of independent groups of dimensionless numbers that are created in this process. The notion of independence is best illustrated by an example which follows.

The following example intends to illustrate how row-rank predicts the effect of scaling constant parameters by varying parameters using a fictitious system. Let $q_1$, $q_2$

and $q_3$ be three constant parameters spanning three dimensions, and these constants are going to be scaled by varying parameters. Also assume the scaling parameters are $q_4$, $p_1$ and $p_2$, where $q_i$'s are constant parameters and $p_i$ are varying parameters. Now, assume $C_S$ has been calculated for the system and is given by Eq. **4.3**,

$$C_S = \begin{bmatrix} n_1 & -1 & 1 \\ n_2 & -1 & 1 \\ n_3 & -1 & 1 \end{bmatrix} \tag{4.3}$$

The matrix $C_S^{(q1)(p2)}$ given in Eq. **4.4** obviously has a row rank equal to 1,

$$C_S^{(q1)(p2)} = \begin{bmatrix} -1 & 1 \\ -1 & 1 \\ -1 & 1 \end{bmatrix} \tag{4.4}$$

The dimensionless parameters corresponding to the parameters $q_1$, $q_2$ and $q_3$ are then determined (Eq. **4.5** ) as,

$$\begin{aligned} \pi_1 &= q_1^1.q_4^{n_1}.p_2^{-1}.p_3^1 \\ \pi_2 &= q_2^1.q_4^{n_2}.p_2^{-1}.p_3^1 \\ \pi_3 &= q_3^1.q_4^{n_3}.p_2^{-1}.p_3^1 \end{aligned} \tag{4.5}$$

where, $n_i$'s are the elements of $C_S^{(q1)(q2)}$, i.e., $C_S^{(q1)(q2)} = \begin{bmatrix} n_1 & n_2 & n_3 \end{bmatrix}^T$. Note that the values of $n_1 \cdots n_3$ are not important because $q_4$ is a constant (raising a constant to arbitrary integer powers produces another constant). It appears in Eq. **4.5** that all three dimensionless parameters are varying. However, they are varying in the same unit dimensional "direction", i.e. all are varying by the same scaling factor. In other words, two of the dimensionless parameters are a constant product of the first dimensionless parameter. Hence, only one varying parameter is actually needed to represent the three parameters in this gain-scheduling problem example.

The previous examples and discussion should illustrate that the lesser the rank of $C_S^{(q1)(p2)}$, the lesser the number of the newly created varying parameters as a result of the transformation, and the lesser the number of gain-scheduling parameters in the dimensionally transformed system. We summarize these results with *Theorem 1* given below.

**Theorem 4-1**: *Let the mathematical model of a system with* $N_p$ *total varying parameters that require gain-scheduling undergoes an arbitrary dimensional transformation. Suppose* $N_{p,S}$ *varying parameters are used for dimensional rescaling of the remaining parameters. Then the number of varying parameters that require gain-scheduling in the dimensionless representation is given exactly by* $N_{\pi,GS}$ (Eq. **4.6)**, *and is defined as:*

$$N_{\pi,GS} = N_p - N_{p,s} + RowRank\ (C_S^{(q1)(p2)}) \qquad \textbf{4.6}$$

*where* $C_S^{(q1)(p2)}$ *is the sub- matrix as defined by* Eq. **4.2**.

**Proof**:. Referring to Table **4-1**, note that the super matrix formed by $\left[\begin{array}{c|c} B_d & A_d \\ \hline I & C_s \end{array}\right]$ is square by construction, so denote the dimension of the square super matrix as $N$. The total number of columns in this super matrix, i.e. the number of variables including signals, varying and non-varying parameters, is given by $N = N_x + N_q + N_p$

where $N_x$, $N_P$, and $N_q$ are the number of signals, varying parameters and non-varying parameters, respectively. By inspection, the number of rows in the super matrix is equal to $N = N_e + N_\pi$. Since there are $N_\pi$ parameters in the dimensionless representation and $N$ in the original representation, the number of variables eliminated by the dimensional transformation is $N_e$ (note: the above is just a matrix restatement of the Buckingham Pi-Theorem).

The number of parameters used in performing the scaling necessary for dimensional transformation is given by $N_{p,s} = N_e - (N_q - N_{q1})$. These parameters exist only as multiplying/dividing factor to the rest of the parameters. However, these scaling factors are time varying, and therefore may create new varying parameters when non-varying parameters are being scaled by the varying parameters. The number of the newly created varying dimensionless parameters are equal to the $RowRank\ (C_S^{(q1)(p2)})$. But, $N_{p,s}$ varying parameters are used to scale other parameters and do not exist independently. Therefore, total gain-scheduling parameters in the new dimensionless representation are, $N_{\pi,GS} = N_p - N_{p,s} + RowRank\ (C_S^{(q1)(p2)})$.

This completes the proof. ■

***Example***: To illustrate the use of the above theorem, we reconsider the gantry system. From before, the system has a total of five parameters, $m_p, m_t, b, L, g$. Three of these variables, $m_p, L, m_t$, are hereafter assumed to be varying significantly and will require gain-scheduling ($N_p = 3$). The remaining two variables are assumed to be constant ( $N_q = 2$ ). Therefore, for the gantry example, the different sub-matrices of Table **4-1** are given by equations Eq. **4.7** - Eq. **4.10**:

$$C_S^{(x)} = \begin{bmatrix} 1/2 & 0 & -1/2 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & -1 & 0 \end{bmatrix} \qquad \mathbf{4.7}$$

$$C_S^{(p1)} = \begin{bmatrix} 0 & -1 & 0 \end{bmatrix} \qquad \mathbf{4.8}$$

$$C_S^{(q1)(q2)} = \begin{bmatrix} -1/2 \end{bmatrix} \qquad \mathbf{4.9}$$

$$C_S^{(q1)(p2)} = \begin{bmatrix} -1 & 1/2 \end{bmatrix} \qquad \mathbf{4.10}$$

Also, since the dimensional unit that spans all the variables in the gantry system is $e = \begin{bmatrix} length & mass & time \end{bmatrix}^T$, then $N_e = 3$. From the given choice of $A_D$ and $B_D$ in Table **4-2** (initially given in chapter 3), $N_{q1} = 1$ (only one non-varying parameter is present in $B_D$).

Table **4-2**: The gantry example

| | $b$ | $m_t$ | $x$ | $\theta$ | $u$ | $t$ | $g$ | $m_p$ | $L$ |
|---|---|---|---|---|---|---|---|---|---|
| $m$ | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| $kg$ | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| $s$ | $-1$ | 0 | 0 | 0 | $-2$ | 1 | $-2$ | 0 | 0 |
| $\pi_1$ | 1 | 0 | 0 | 0 | 0 | 0 | $-1/2$ | $-1$ | $1/2$ |
| $\pi_2$ | 0 | 1 | 0 | 0 | 0 | 0 | 0 | $-1$ | 0 |
| $\pi_3$ | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | $-1$ |
| $\pi_4$ | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| $\pi_5$ | 0 | 0 | 0 | 0 | 1 | 0 | $-1$ | $-1$ | 0 |
| $\pi_6$ | 0 | 0 | 0 | 0 | 0 | 1 | $1/2$ | 0 | $-1/2$ |

Hence, $N_{p,s}$ is calculated as: $N_{p,s} = N_e - (N_q - N_{q1}) = 3 - (2-1) = 2$. Also, $RowRank\ (C_S^{(q1)(p2)}) = 1$. Therefore, as a result of the transformation the number of gain-scheduling parameters in the new representation is calculated as: $N_{\pi,GS} = N_p - N_{p,s} + RowRank\ (C_S^{(q1)(p2)}) = 3 - 2 + 1 = 2$. Thus, the dimensionally transformed gantry system results in two gain-scheduling parameters, which is one less than the original representation.

The previous theorem allows consideration of special cases that are revealing of the potential benefit of the method. The gain-scheduling parameters can be reduced by a maximum value equal to $N_e$, which is the size of the basis dimensional vector. A dynamic system, for example, usually has unit dimensional vector of size $N_e = 3$ corresponding to length, mass, and time, e.g. $e = [length \quad mass \quad time]^T$. So in general, one can expect to eliminate at most 3 gain scheduled variables from a typical dynamic representation. While elimination of 3 parameters seems like only a modest improvement, if each variable requires 10 levels of consideration, Eq. **4.6** then suggests that the proposed method might then reduce problem complexity by a factor of 1000.

In the trivial case where the dynamic representation is already dimensionless ($N_e = 0$), clearly there can be no reduction in the number of gain-scheduled parameters, and there may be potentially negative effects in attempting to scale the system further.

In the case where all the problem parameters are being gain scheduled ($N_q = 0$), the number of gain-scheduling parameters of new representation is: $N_{\pi,GS} = N_p - N_e$. A subset of this case is the very special situation where, $N_p = N_e$. In this case, all the $N_p$

parameters are used to scale the signals alone, and there is no gain-scheduling parameter left in the dimensionless domain. This results in an LTI system in the dimensionless domain. In other words, there exist classes of LPV systems that can be transformed to LTI systems under dimensional transformation. This conversion of the LPV system to an LTI system through the dimensional transformation results in an *implicit* gain-scheduling problem as shown in Figure **4-3**, where gain scheduling solely occurs in the transformation process from one unit system to another.

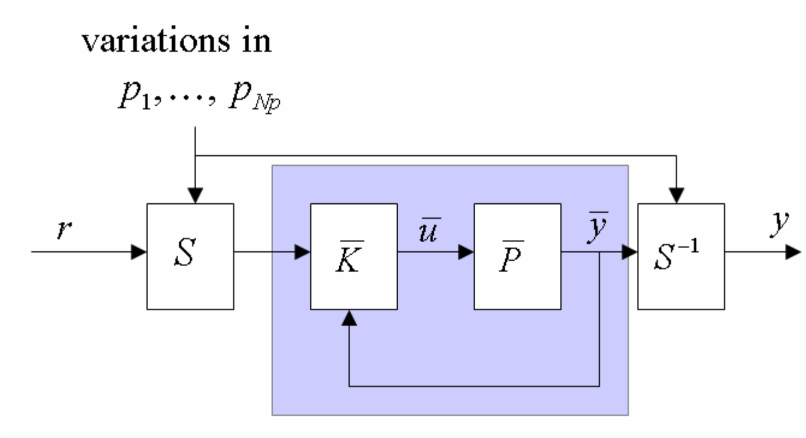



Figure **4-3**: In certain cases, the scaling parameters alone fully account for varying parameters of the system, resulting in a LTI plant and LTI controller design in the dimensionless domain (shaded) even though the classical system is LTV and would seem to require gain-scheduling.

This conversion of an LPV gain-scheduled system to an LTI system can be illustrated using the simple example of mass-spring system whose equation of motion is given by Eq. **3.11**.

$$m\ddot{x} + kx = F \qquad \textbf{4.11}$$

Assume that both parameters $m$ and $k$ are varying, therefore the dynamic model is an LPV system and requires gain-scheduling for a controller synthesis. Defining the new dimensionless variables and time as Eq. **3.12**, the dimensionless form of the original system is given by Eq. **3.13**.

$$\bar{x} = x \cdot \frac{k}{mg}; \quad \bar{F} = F \cdot \frac{1}{mg}; \quad \tau = t \cdot \sqrt{\frac{k}{m}} \qquad \textbf{4.12}$$

$$\bar{x}'' + \bar{x} = \bar{F} \qquad \textbf{4.13}$$

The system in Eq. **3.13** is parameter-free, and therefore clearly an LTI system in the dimensionless domain. It would therefore require no gain scheduling in this domain. Although the two system representation, Eq. **3.11** and Eq. **3.13**, look different in form they are equivalent from a control view point. Also note that both are marginally stable. An input that (de)stabilize the representation in Eq. **3.11** will also (de)stabilize Eq. **3.13** and vice versa. For example, in Eq. **3.13** the system becomes unstable if the dimensionless input force has a dimensionless frequency $\bar{\omega} = 1.0$. However, the dimensionless frequency is related to the dimensional frequency using the time scaling as Eq. **3.14**.

$$\omega = \bar{\omega} \cdot \sqrt{\frac{k}{m}} \qquad \textbf{4.14}$$

Therefore, an input frequency that is a resonance frequency for the model in Eq. **3.13** is also a resonance frequency for the model in Eq. **3.11** and vice versa.

### 4.5 The invariant control space

The proposed control technique for gain-scheduling is based on a re-parameterization of the system representation into an equivalent but dimensionless formulation. The dimensionless representation and controller for this system can then be scheduled with respect to fewer parameters, as shown in Figure **4-4**.

Because the controller gains in the dimensionless case are dependent only on $\pi_1$ and $\pi_2$, it is possible that variations in the parameters $m_p$, $L$ and $m_t$ may not affect the controller. Specifically, if $\pi_1$ and $\pi_2$ are constant despite these changes, no controller gain-scheduling will be necessary. While such type of coupling is unlikely for systems with arbitrary parameter variation, it is common for real engineering systems that parameter variations occur in a coupled manner, for instance an object with increased mass nearly always has increased rotational inertias.

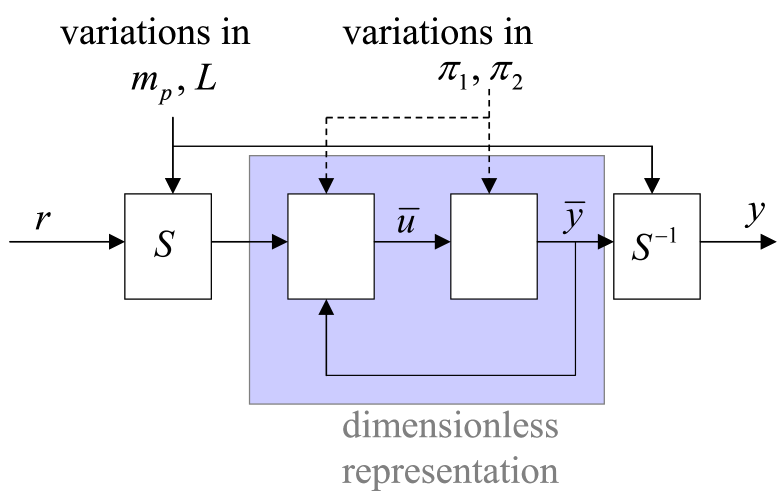



Figure **4-4**: By eliminating one parameter in the process of rescaling to and from a dimensionless representation, the resulting dimensionless control synthesis (shaded) will schedule only with the two new parameters.

A direct consequence of this reduction in parameter space is that there is an invariant control space. That is, all the plants in this invariant space can be controlled by a single dimensionless controller. To demonstrate this, the same controllers $K$ and $\overline{K}$ has been used to control a new plant that has different $m_p$ , $L$ and $m_t$ from the previous case but the same $\pi_1$ and $\pi_2$, more specifically, $m_p = 0.784kg$, $m_t = 4.84kg$, $L = 4.976m$ .

The performance of the dimensionless controller, as shown in Figure **4-5**, shows that a single controller can control plants with different physical parameters but the same dimensionless parameters.

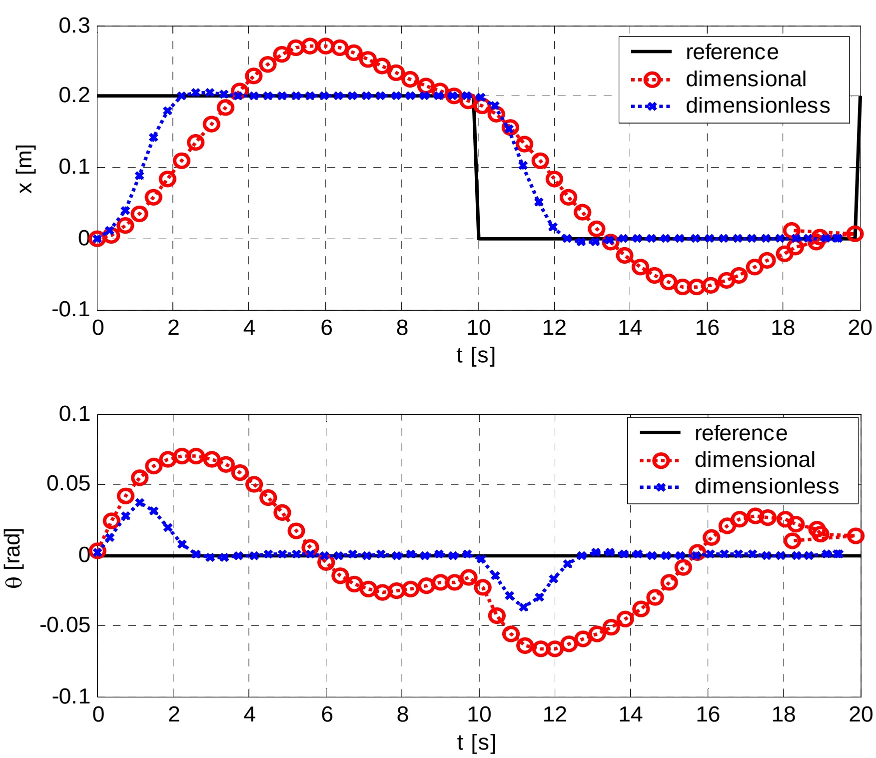



Figure **4-5**: Use of a single dimensionless controller for plants in the invariant space in relation to the dimensionless parameters.

Because typical controller synthesis techniques do not account for this invariance, a significant degradation in performance is seen when comparing the two. The use of a single dimensionless controller for plants in the invariant space shows results consistent with the original design. However, the classical representation (circles) shows a severe loss in performance under the same parametric changes. Note that the change of the parameters in this example is artificially constrained to illustrate a controller's insensitivity to parameter variation, e.g. when parameters couple in the most ideal way. A control designer working in the classical domain, unaware of the control insensitivity to certain types of parameter coupling, might assume gain-scheduling is necessary. But this is not the case in the dimensionless domain: gain-scheduling is not necessary in the dimensionless controller.

## 4.6 A Search Method for Optimum Scaling Matrices

In section **4.4**, a technique that yields guaranteed reduction of parametric space for gain scheduling for a given scaling matrix was developed. However, the choice of the scaling matrix remained a user-choice, and this choice directly affects the optimality of the parametric minimization. The central idea of this chapter is to optimize all possible scaling matrices to find the minimum number of scheduling parameters possible for a given system. This involves the search for all optimum scaling matrices that yield a transformed-system with the minimum number of scheduling parameters. Since the optimization problem is basically discrete and finite dimensional optimization problem, it is solved using a combinatorial optimization technique.

### 4.6.1 Combinatorial Outcomes

The combinatorial outcomes, simply called combination, "$n$ choose $k$" is the number of ways of $k$ unordered outcomes from $n$ possibilities and is represented as ${}_nC_k$. The mathematical definition of combination is given by Eq. **3.1**.

$$ {}_nC_k \equiv \binom{n}{k} \equiv \frac{n!}{k!(n-k)!}, \quad \text{where, } ! \equiv \textit{factorial} \qquad \textbf{4.15} $$

***Example***: Professor Brennan has five veteran graduate students (B, H, R, S, V) and he wants to send a team of three graduate students to the 2006 ACC conference. The total number of ways he can choose the team is:

$$ {}_5C_3 = \binom{5}{3} = \frac{5!}{3!(5-3)!} = 10 . $$

In the dimensional transformation method, the elements of the scaling matrix $A_D$ are dependent on the choice of the scaling parameters (also traditionally known as repeating parameters) and the unit systems. The size (dimension) of $A_D$ is equal to the number of basis dimensional units, $N_e$, used to describe the dynamic system. Therefore, the total number of ways of choosing $N_e$ scaling parameters out of $N$ parameters is given by, Eq. **3.2**.

$$ N_T = {}_NC_{N_e} = \binom{N}{N_e} = \frac{N!}{N_e!(N-N_e)!} \qquad \textbf{4.16} $$

However, the actual number of admissible scaling matrices, $N_A$, is not always equal to $N_T$ because any combination of scaling parameters that yield a singular matrix $A_D$ is not admissible. If the total number of ways of getting a singular scaling matrix is $N_S$, then the number of admissible scaling matrices is given by Eq. **3.3** and any optimization search is confined to these admissible scaling matrices only.

$$N_A = N_T - N_S = \frac{N!}{N_e!\left(N - N_e\right)!} - N_S \qquad \textbf{4.17}$$

### 4.6.2 The Combinatorial Optimization

The combinatorial optimization here is performed by evaluating the value of a given objective function at each discrete combinatorial outcome. Since the objective here is to minimize the number of gain scheduling parameters, the objective function, initially defined in Eq. **4.6**, is $N_{\pi,GS} = N_p - N_{p,s} + RowRank\ (C_S^{(q1)(p2)})$. Therefore, the optimum solution is one that minimizes the objective function. The optimization problem is therefore given by Eq. **3.10**. The algorithm to solve this optimization problem assumes the transformation matrices are as defined in Table **4-1** and is given in Table **3-1**.

$$\begin{aligned}
&\textit{find all } A_D^{opt} \in A_D \\
&\textit{such that} \\
&\qquad N_{\pi,GS} = N_p - N_{p,s} + RowRank\ (C_S^{(q1)(p2)}) \\
&\qquad \textit{is minimized}
\end{aligned} \qquad \textbf{4.18}$$

The computer code to solve this optimization is written in MATLAB and can be found in Appendix **A.1**.

Table **4-3**: The combinatorial optimization algorithm

| |
|---|
| 1. determine $N_T = {}_N C_{N_e}$ |
| 2. generate $A_D(i), \quad i = 1 \ldots N_T$ |
| 3. set $i = 0$, $N_{\pi,GS}^{opt} = N_p$ and $j = 0$ |
| 4. $i = i + 1$ |
| 5. if $A_D(i)$ is singular → go to step 4 |
| else calculate $N_{\pi,GS} = N_p - N_{p,s} + RowRank\ (C_S^{(q1)(p2)})$ |
| 6. if $N_{\pi,GS} > N_{\pi,GS}^{opt}$ → go to step 4 |
| else if $N_{\pi,GS} = N_{\pi,GS}^{opt}$ |
| set $j = j + 1$ |
| calculate $A_D^{opt}(j) = A_D(i)$ |
| else, set $j = 1$ and $A_D^{opt} = [\ ]$ |
| calculate $A_D^{opt}(j) = A_D(i)$ |
| 7. if $i < N_T$ → go to step 4 |
| else, STOP. |

***Example***: The gantry problem has been solved using this algorithm and the optimum solution is: $N_{\pi,GS}^{opt} = 2$. The algorithm found seven ways of scaling the system that all give the optimum, minimum gain-scheduling parameter representation. The code also automatically generates the $\pi$-parameters for each of the optimum solution and the first three results are presented symbolically in Table **4-4** . Note that in all the transformed variables there are two gain-scheduling parameters. The code and the supporting functions are listed in Appendix **A.1**.

Table **4-4**: Optimum transformation results

| | | | |
|---|---|---|---|
| $\pi_1$ | $L \cdot \frac{b^2}{m_t^2 g}$ | $L \cdot \frac{b^2}{m_p^2 g}$ | $m_t \cdot \frac{1}{b}\sqrt{\frac{g}{L}}$ |
| $\pi_2$ | $\frac{m_p}{m_t}$ | $\frac{m_t}{m_p}$ | $m_p \cdot \frac{1}{b}\sqrt{\frac{g}{L}}$ |
| $\pi_3$ | $x \cdot \frac{b^2}{m_t^2 g}$ | $x \cdot \frac{b^2}{m_p^2 g}$ | $x \cdot \frac{1}{L}$ |
| $\pi_4$ | $\theta$ | $\theta$ | $\theta$ |
| $\pi_5$ | $u \cdot \frac{1}{m_t g}$ | $u \cdot \frac{1}{m_p g}$ | $u \cdot \frac{1}{b\sqrt{gL}}$ |
| $\pi_6$ | $t \cdot \frac{b}{m_t}$ | $t \cdot \frac{b}{m_p}$ | $t \cdot \sqrt{\frac{g}{L}}$ |

## 4.7 Summary

The reduction in parameters has a profound impact on the design of gain scheduling control systems because it implies significant (exponentially smaller) problem simplification and controller representation. A method of gain-scheduling parametric space minimization was shown using dimensional transformation. Higher degree of gain-scheduling parameter minimization is achieved by careful choice of the scaling and scaled parameters during the dimensional transformation. It was also shown that under special cases, an LPV system may be converted to an LTI system during such transformation.

Equivalence between dimensionless and dimensioned controllers was demonstrated using a gantry system. The gantry results illustrate that, by carefully choosing certain gain-scheduling variables for dimensional transformations, gain-scheduling parameters can be reduced by mapping the system to a lower parameter space (i.e., from three to two, in this case) in the dimensionless system representation.

The optimum system representation problem was formulated as a combinatorial optimization problem and algorithm was developed to solve the problem.

### 4.8 References


[1] F. Bruzelius, "LPV-Based Gain Scheduling, An $H_\infty$–LMI Approach," in *Signals and Systems*. Göteborg: Chalmers University of Technology, 2002, pp. 112.

[2] D. Bates and I. Postlethwaite, *Robust Multivariable Control of Aerospace Systems*: Delft University Press, 2002.

[3] T. Shimomura, "Hybrid Control of Gain-Scheduling and Switching: A Design Example of Aircraft Control," American Control Conference, 2003.

[4] W. J. Rugh, "Analytical Framework for Gain Scheduling," in *IEEE Control Systems Magazine*, vol. 11, 1991, pp. 79-84.

[5] D. J. Leith and W. E. Leithead, "On Formulating Nonlinear Dynamics in LPV Form," Proceedings of the 39th IEEE Conference on Decision and Control, 2000.

[6] J. W. Choi and P. G. Park, "Guaranteed Cost LPV Output-Feedback Controller Design for Nonlinear Systems," Proceedings of the 2001 IEEE International Symposium on Industrial Electronics, 2001.

[7] C. H. Lee and M. J. Chung, "Gain-Scheduled State Feedback Control Design Technique for Flight Vehicles," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 37, pp. 173-182, 2001.

[8] S. J. Fedigan, C. R. Knospe, and R. D. Williams, "Gain-Scheduled Control for Substructure Properties," Proceedings of the 1998 American Control Conference, 1998.

[9] G. Corriga, A. Giua, and G. Usai, "An Implicit Gain-Scheduling Controller for Cranes," *IEEE Transactions on Control Sysytems Technology*, vol. 6, pp. 15-20, 1998.

[10] S. Tavakoli and M. Tavakoli, "Optimal Tuning of PID Controllers for First Order plus Time Delay Models Using Dimensional Analysis," The 4th IEEE International Conference on Control and Automation, Canada, 2003.

[11] K. J. Astrom, C. C. Hang, P. Persson, and W. K. Ho, "Towards Intelligent PID Control," *Automatica*, vol. 28, pp. 1-9, 1992.

[12] F. Wang and V. Balakrishnan, "Improved Stability Analysis and Gain-Scheduled Controller Synthesis for Parameter-Dependent Systems," *IEEE Transactions on Automatic Control*, vol. 47, pp. 720-734, 2002.

## Chapter 5

# Robust Control and Parametric Uncertainty

### 5.1 Introduction

This chapter discusses a technique of reducing the parametric uncertainty block size in an LFT framework of robust control problems. The goal is to reduces conservativeness and complexity of the robust control synthesis [1-3]. Using the return difference matrix determinant condition for robust stability, it is shown in this chapter that the dimensionless representation results in larger allowable parametric perturbation compared to the dimensional approach. A side-by-side comparison of the dimensional and dimensionless control synthesis is performed to numerically illustrate the advantages of the proposed technique. Three potential drawbacks of the proposed method that are problem dependent are identified and potential remedies are presented.

Robust control design is based on the idea that the plant can be described as a nominal system with additional unmodeled dynamics such as nonlinearities, immeasurable noise, parametric uncertainties or modeling errors, etc. Robust control systems are intended to be insensitive to differences between the actual system and the model of the system which is used to design the controller assuming particular bounds in the uncertainty. There are many origins for the model/plant mismatch [4, 5]:

1. Real systems are generally nonlinear.

2. There are always parameters in the linear model which are only known approximately.
3. The parameters in the linear model may vary due to nonlinearities or changes in the operating conditions.
4. There are always imperfections in the measurement / actuation systems.
5. Even when a detailed model is available, one may choose to work with a simpler model (low-order) nominal model and represent the neglected dynamics as uncertainty.
6. The controller implemented may differ from the one obtained by solving the synthesis problem. In this case one may include uncertainty to allow for controller order reduction and implementation inaccuracies.

The various sources of model uncertainty mentioned above are generally grouped into two main classes [4, 5].

1. ***Parametric uncertainty***: Here the structure of the model (including the order) is known, but some of the parameters are uncertain. Therefore, the system can be modeled exactly except that the parameters of the model are not precisely known. For example consider a mass-spring system and assume that the system can be modeled as a linear system, $m\ddot{x} + kx = F$. One cannot claim that the system is exactly known without being able to identify the exact values of the parameters $m$ and $k$. If $m$ and $k$ are known partially, i.e., they are known to lie within a known range, the model is said to exhibit parametric uncertainty.
2. ***Unmodelled dynamics uncertainty***: Here, the model is in a mismatched condition because of missing dynamics, usually at high frequencies. This uncertainty often

results from one's choice to work with simpler models for practical reasons. All models of real system will contain this source of uncertainty to some degree.

The core of the robust control design paradigm is based on notion of a family of plants which are generally represented as a nominal plant and all perturbations from the nominal, grouped as an uncertainty set. The robust controller synthesis problem, therefore, seeks to find a single controller that is guaranteed to stabilize all the plants in the family.

## 5.2 Literature Review

Thus far, two types of stability conditions have been reported in the study of robust control techniques: those based on the Small Gain Theorem, which result in circle conditions in the complex plane, and those based on Passivity Theorem. The two approaches are conceptually different and have each led to important results.

$H_\infty$ control synthesis based on small gain theorem was introduced by Zames [6], where the problem of $H_\infty$ control is cast as an optimization problem subject to constraints in the frequency domain. Since then, a very significant body of research has developed advanced techniques on both analysis and synthesis of the robust control systems.

One of the most important steps in robust control design in all frameworks is the representation of the system in a Linear Fractional Transformation (LFT) form. This form, usually represented in short hand as $P-\Delta$, is a separate system description of the nominal input-output system $P$ and the uncertainty input-output system $\Delta$. For practical problems, the formulation of a $P-\Delta$ model which accurately characterizes realistic system uncertainties is critical because the robustness results of any controller design depend directly on the uncertainty model used in the analysis or design. An overview of such modeling is presented in Belcastro [7].

Parametric uncertainties usually originate from inexact estimation or measurement of the parameters in a model, change of operating conditions, etc. whereas, un-modeled dynamics uncertainties are usually the result of simplification of the model to reduce the system order. While there is a clear difference between the two types of uncertainty, some researchers have shown that they can be unified into one. For example, Fu [8] has shown that the robust stability problem for linear system with both parametric and nonparametric uncertainties can be equivalently represented as a robust stability problem with parametric uncertainty only.

Many researchers have focused on the integration of robust stability and robust performance problems into a single robust stability problem. In [9], Sideris showed that robust performance in linear feedback systems with: 1) parametric model uncertainty and 2) robust stability requirement under combined parametric model uncertainty and unmodeled dynamics are reduced to an equivalent single problem of analyzing robust stability with respect to the uncertain parameters. The stability condition is checked by computing the multivariable stability margin (MSM). Collado and Rojas [10] has proposed a technique for robust stability based on the field of values of a matrix to determine the stability of state-space models with parametric uncertainty.

Many practical problems have a structured uncertainty description, i.e., the uncertainty block have diagonal elements or blocks. Any control design that does not take into account this extra information about uncertainty structure may produce a more conservative controller synthesis. In 1982 both Doyle [11] and Safonov [12] independently developed a robustness measure that takes into account the structure of an uncertainty description of many practical problems. The former introduced the structured singular value – $\mu$. The robust control synthesis procedure that is based on this structured singular value is called $\mu$-synthesis. The basic idea in both cases is to use an extension of the Nyquist theorem to MIMO systems.

The two basic methods for $\mu$-synthesis developed thus far are: the $D-K$ iteration by Doyle [13] and the $\mu-K$ iteration by Lin et al. [14].

Many researchers show the existence of alternative approaches to the structured singular value formulations. For example, Basker et. al. [15] propose a new and simple approach to the problem of finding robust stability margins for SISO systems with complex parametric uncertainties based on the critical direction theory. Sidersis and Sanchez Pena [16] considered the problem of robust stability in feedback control systems with real uncertain parameters and unmodelled dynamics by defining a robust margin $r_m$ and also developed algorithm to calculate $r_m$ and the closely related multivariable stability margin $k_m$ (the inverse of $\mu$).

Although the structured singular value, $\mu$, has been a useful tool for matrix perturbation problems, its computation is very difficult. The reason so many have focused on simplification of uncertainty representation is because problem complexity and conservativeness increases with size of the uncertainty block. Fan et. al. [2] discussed the exponential increase in computational complexity with the number of uncertain parameters/blocks. Current computational methods of the structural singular value, $\mu$, are often limited to calculating its upper and lower bounds. Upper bounds give conservative estimates of the sizes of allowable perturbations. For uncertainty block size $\leq 3$, the actual $\mu$ can be computed [1, 2, 17]. However, for block size > $3$, $\mu$ cannot be computed exactly, and the gap between the upper and lower bounds can be arbitrarily large resulting in a more conservative controller synthesis [2, 17]. Therefore, a technique that alleviates this limitation in any way may result in a significantly less conservative result. This is the objective of this chapter.

Others have considered robust stability/performance for sampled-data systems. To mention few: Ito [18] used a scaled small gain conditions to directly derive the robust performance of sampled-data systems, where the uncertainty is restricted to be finite dimensional linear time invariant (FDLTI) and the scaling is allowed to be frequency dependent. Shi et al. [19] considered the design of output feedback control for a class of sampled-data systems with parametric uncertainties, where both the problems of robust stabilization and robust $H_\infty$ performance were tackled by converting to $H_\infty$ synthesis for sampled-data systems without parametric uncertainties.

Another interesting problem from robust control implementation perspective is that in many applications the robust controllers are not implemented as designed. For practical reasons, the implementation is usually done using a reduced order controller. Sometimes the reduction in order may just require only minor adjustments (tuning) of the coefficients of the controller for better performance. The tuning of controller coefficients may deteriorate the controller effectiveness in robustly stabilizing the system. This potential problem is pointed by Keel and Bhattacharyya [20]. Many researchers have addressed this fragility problem through the deign of resilient or non-fragile robust synthesis methods for certain types of problem [21-24]. The proposed technique in this chapter involves the input/output and time scaling of the controller using selected parameters that sometimes are uncertain.

## 5.3 Robust Control

The robust control design approach becomes necessary when there are system uncertainties such as parametric variations or uncertainties, un-modeled/neglected dynamics, disturbances, etc. The basic idea in this approach is that, for a given nominal system description, $P$, and a description of the uncertainties $\Delta$, one can design a controller that stabilizes and meets all performance requirements for all possible plants and disturbances within the uncertainty description. Current robust control synthesis methods require the formulation of the plant and uncertainty descriptions in a form called linear fractional transformation (LFT) as shown in Figure **5-1**. This form is usually represented in short hand as $P-\Delta$, is a structure where the uncertainty is replaced by a input-output disturbance pairs that are related by the uncertainty block $\Delta$.

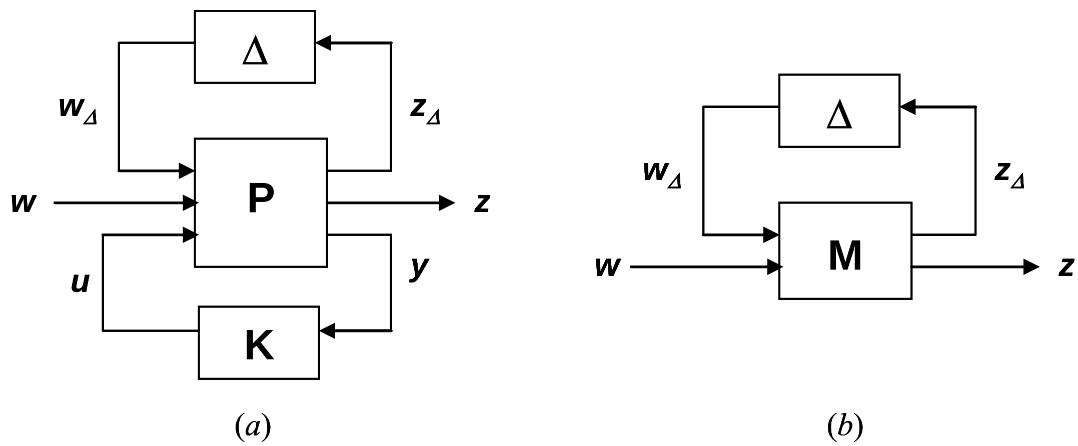



Figure **5-1**: A common robust control setup. (*a*) Block diagram of nominal plant–controller–uncertainty interconnection structure, (*b*) Block diagram of closed loop system–uncertainty interconnection structure

In Figure **5-1(**$b$**)** assume that the system $M(s)$ is nominally stable and the perturbations $\Delta(s)$ are stable. If these conditions are satisfied, then the $M-\Delta$ structure is stable [4] for all perturbations $\Delta$ satisfying $\|\Delta\|_\infty \le 1$ if and only if

$$\bar{\sigma}\big(M(j\omega)\big) < 1 \quad \forall\omega \quad \Leftrightarrow \quad \|M\|_\infty < 1 \qquad \textbf{5.1}$$

where, $\|.\|_\infty$ is the $H_\infty$ norm, also called the $L_2 - gain$, of the system. The condition for robust stability (RS) given in Eq. **5.1** can also be rewritten as

$$RS \quad \Leftrightarrow \quad \bar{\sigma}\big(M(j\omega)\big)\bar{\sigma}\big(\Delta(j\omega)\big) < 1 \quad \forall\omega, \forall\Delta \qquad \textbf{5.2}$$

In the next section an example is presented to illustrate the differences between the dimensionless and dimensional forms of controller synthesis.

## 5.4 Example: The 1990 ACC Benchmark Problem

The 1990 ACC benchmark problem [25] is presented in this chapter to illustrate the benefit of the parametric uncertainty block reduction using dimensional transformation. The system consists of a coupled two-mass and a spring system without damping as shown in Figure **5-2**. It is input-to-output non-collocated because the actuator is acting on $m_1$ while only the position of $m_2$ is measured. This system is commonly used in the robust control community to test and validate new techniques.

To demonstrate the benefit of dimensional transformation, two different controllers are designed in both the dimensioned and dimensionless domains and the performance of each controller is compared.

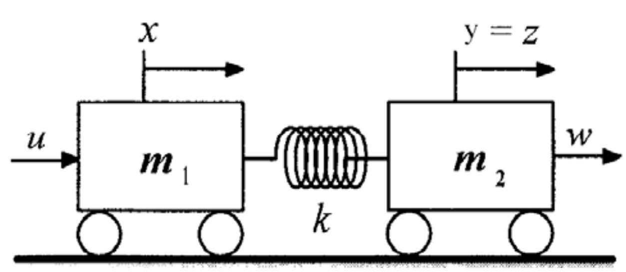



Figure **5-2**: The 1990 ACC benchmark problem

### 5.4.1 Modeling of the parametric uncertain system

The equation of motion of the system shown in Figure **5-2** is, in the classical form, given by Eq. **5.3.**

$$\begin{aligned} m_1\ddot{x} + kx - ky &= u \\ m_2\ddot{y} - kx + ky &= w \end{aligned} \qquad \textbf{5.3}$$

All three parameters in Eq. **5.3** are uncertain, i.e., $m_1 = m_{10} + \delta_{m1}$, $m_2 = m_{20} + \delta_{m2}$, $k = k_0 + \delta_k$, where $m_{10}$, $m_{20}$, $k_0$ are the nominal parameters and $\delta_i$'s are the perturbations from the nominal values and are assumed to be bounded, for instance in this example, it's assumed that $\|\delta_i\| \le 0.2$. The model of Eq. **5.3** can be represented in the classical form and parametric uncertainty form as shown in Figure **5-3** (*a*) and Figure **5-3** (*b*), respectively. However, for robust control synthesis, the uncertainty model representation is used because it is easy to convert it to the LFT form of the system. As can be seen from Figure **5-3** (*b*), the system has a diagonal uncertainty block of size three.

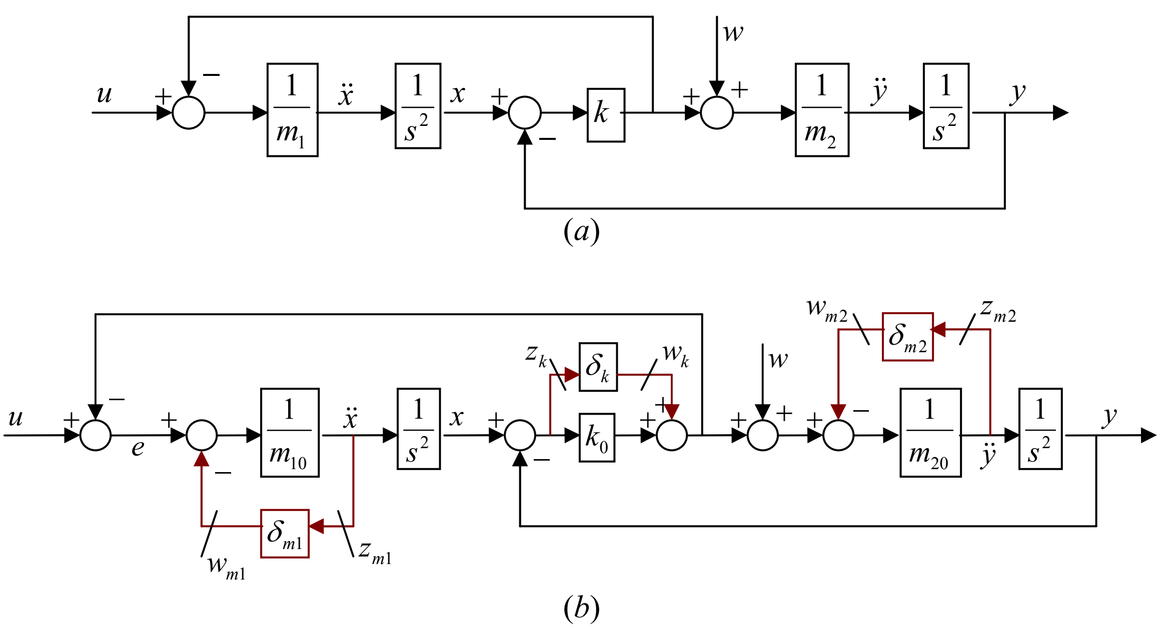



Figure **5-3**: Dimensional model: (*a*) classical model, (*b*) uncertainty model

The uncertainty system is represented by the input-output pair $\mathbf{w} - \mathbf{z}$ with $\mathbf{w} = [w_k \quad w_{m1} \quad w_{m2}]^T$ and $\mathbf{z} = [z_k \quad z_{m1} \quad z_{m2}]^T$ that are related through, Eq. **5.4**.

$$\begin{bmatrix} w_k \\ w_{m1} \\ w_{m2} \end{bmatrix} = \begin{bmatrix} \delta_k & 0 & 0 \\ 0 & \delta_{m1} & 0 \\ 0 & 0 & \delta_{m2} \end{bmatrix} \begin{bmatrix} z_k \\ z_{m1} \\ z_{m2} \end{bmatrix} \qquad \textbf{5.4}$$

### 5.4.2 The dimensionless representation of the system

The above system is now transformed to dimensionless representation (refer to Chapter **3** for details of the transformation procedure). The scaling parameters are chosen to be $m_1$, $k$ and $g$ and the matrix form is given in Table **5-1**. The resulting dimensionless variables are:

$$\pi_1 = \frac{m_2}{m_1}, \qquad \pi_2 = \frac{kx}{m_1 g} = \bar{x}, \qquad \pi_3 = \frac{ky}{m_1 g} = \bar{y},$$
$$\pi_4 = \frac{u}{m_1 g} = \bar{u}, \quad \pi_5 = \frac{w}{m_1 g} = \bar{w}, \quad \pi_6 = t\sqrt{k/m_1} = \tau \qquad \textbf{5.5}$$

Table **5-1**: The dimensional transformation process

| | $m_2$ | $x$ | $y$ | $u$ | $w$ | $t$ | $m_1$ | $k$ | $g$ |
|---|---|---|---|---|---|---|---|---|---|
| $m$ | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |
| $kg$ | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| $s$ | 0 | 0 | 0 | $-2$ | $-2$ | 1 | 0 | $-2$ | $-2$ |
| $\pi_1$ | 1 | 0 | 0 | 0 | 0 | 0 | $-1$ | 0 | 0 |
| $\pi_2$ | 0 | 1 | 0 | 0 | 0 | 0 | $-1$ | 1 | $-1$ |
| $\pi_3$ | 0 | 0 | 1 | 0 | 0 | 0 | $-1$ | 1 | $-1$ |
| $\pi_4$ | 0 | 0 | 0 | 1 | 0 | 0 | $-1$ | 0 | $-1$ |
| $\pi_5$ | 0 | 0 | 0 | 0 | 1 | 0 | $-1$ | 0 | $-1$ |
| $\pi_6$ | 0 | 0 | 0 | 0 | 0 | 1 | $-1/2$ | $1/2$ | 0 |

Based on the transformation given Eq. **5.5**, the system equation (Eq. **5.3**) in dimensionless form is given by Eq. **5.6**,

$$\begin{aligned} \bar{x}'' + \bar{x} - \bar{y} &= \bar{u} \\ \pi_1 \bar{y}'' - \bar{x} + \bar{y} &= \bar{w} \end{aligned} \qquad \textbf{5.6}$$

Where, $(') \equiv \dfrac{d}{d\tau} = \sqrt{m_1/k}\,\dfrac{d}{dt}$, and the bars on the top of the variables represent the dimensionless (normalized) representation of the variables. Figure **5-4** is the dimensionless representation of the system shown in Figure **5-3**. As can be seen in Figure **5-4** (*b*), the system has only one uncertainty block.

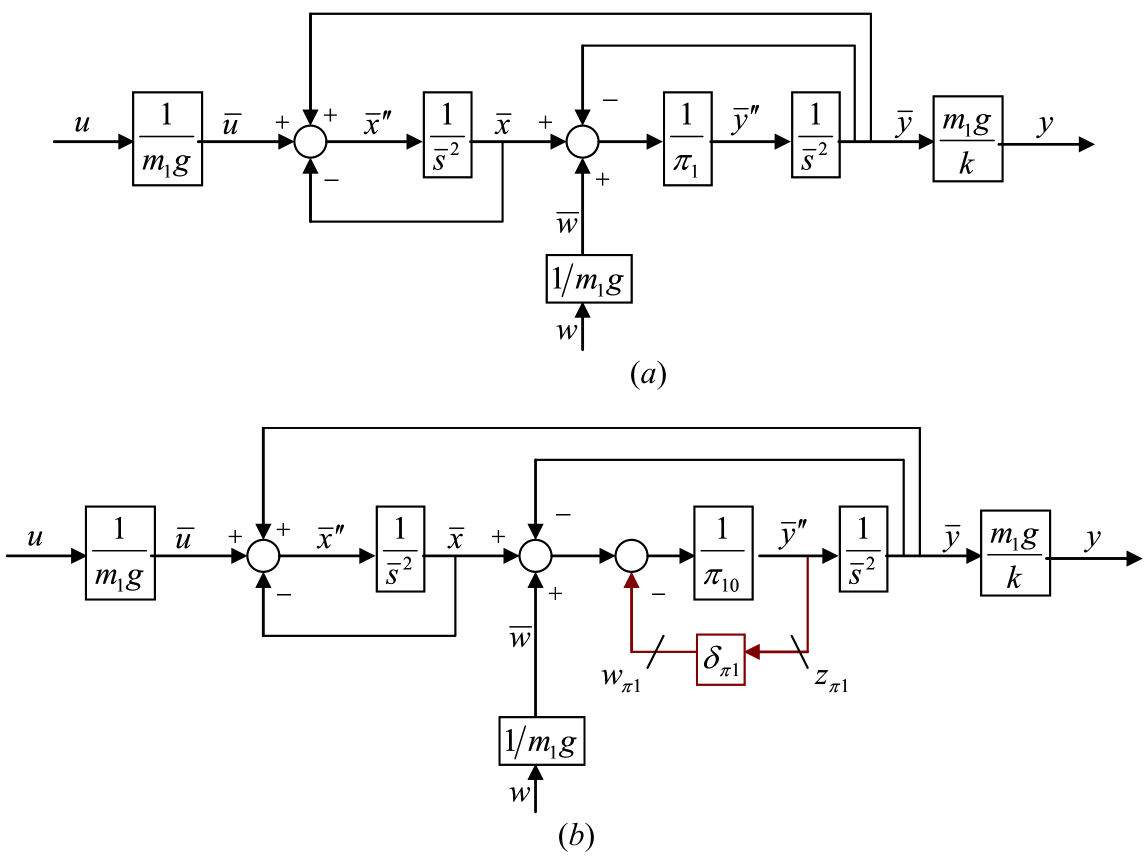



Figure **5-4**: Model in dimensionless form: (*a*) classical model, (*b*) uncertainty model

Hence the parametric uncertainty block size has been reduced from three in the dimensional system to only one in the dimensionless system as a result of the dimensional transformation. Note that there are additional sources of uncertainty – namely those affecting scaling – that will be addressed shortly. In the meantime, the benefit of this uncertainty reduction is explored through a comparison of the numerical results of the two system representations.

### 5.4.3 Simulation Result

To show a comparison in performance of the two system representations, controllers are synthesized for each using the $H_\infty$-synthesis toolbox of MATLAB. The performance specs are chosen to be the same for both cases, i.e., 1) robust stability for all values of $|\delta_i| \leq 0.2$, 2) for the nominal system in response to an impulse disturbance acting at the second mass, the peak control action and the settling time on the displacement of the second mass should satisfy $|u| \leq 1$ and $t_s \sim 15$ sec, respectively.

The nominal parameters used are: $m_{10} = m_{20} = 1.0kg$, $k_0 = 1.0N/m$. The problem is cast as a robust stability/performance problem. Performance weights are selectively imposed on the displacement of the second mass and the control action, Eq. **5.7**, to force both signals $z_y$ and $z_u$ to be within the required values with respect to the disturbance input $w$.

$$\begin{bmatrix} w_k \\ w_{m1} \\ w_{m2} \\ w \end{bmatrix} = \begin{bmatrix} \delta_k & 0 & 0 & 0 & 0 \\ 0 & \delta_{m1} & 0 & 0 & 0 \\ 0 & 0 & \delta_{m2} & 0 & 0 \\ 0 & 0 & 0 & \delta_y & \delta_u \end{bmatrix} \begin{bmatrix} z_k \\ z_{m1} \\ z_{m2} \\ z_y \\ z_u \end{bmatrix} \qquad \mathbf{5.7}$$

The nominal parameter $\pi_{10}$, and the uncertainty bound $\delta_{\pi 1}$ of the dimensionless model are determined from the uncertainties and nominal parameters of the dimensional model, Eq. **5.8** . The dimensionless representation, while having a lower number of uncertain blocks, results in a larger bound than each of the uncertainties in the dimensional representation.

$$\begin{aligned} &\pi_{10} = m_{10} / m_{20} = 1.0/1.0 = 1.0 \\ &\pi_{1\min} = m_{1\min} / m_{2\max} = 0.8/1.2 = 0.67 \\ &\pi_{1\max} = m_{1\max} / m_{2\min} = 1.2/0.8 = 1.5 \\ &\Rightarrow \delta_{\pi 1} = \max\left(\left|\pi_{10} - \pi_{1\min}\right|, \left|\pi_{10} - \pi_{1\max}\right|\right) = 0.5 \end{aligned} \qquad \mathbf{5.8}$$

Similarly, the robust stability/performance setup for the dimensionless representation has the uncertainty block and its input-output pairs, Eq. **5.9**, to force both signals $\bar{z}_{\bar{y}}$ and $\bar{z}_{\bar{u}}$ to be within the required values with respect to the disturbance input $\bar{w}$ (See Appendix **A.2** for the MATLAB code).

$$\begin{bmatrix} \bar{w}_{\pi 1} \\ \bar{w} \end{bmatrix} = \begin{bmatrix} \delta_{\pi 1} & 0 & 0 \\ 0 & \delta_y & \delta_u \end{bmatrix} \begin{bmatrix} z_{\pi 1} \\ \bar{z}_{\bar{y}} \\ \bar{z}_{\bar{u}} \end{bmatrix} \qquad \mathbf{5.9}$$

The numerical results show that the performance specifications were more easily achieved when using the dimensionless representation compared to the dimensional representation. The results of simulation are given below (Figure **5-5** and Figure **5-6**). The first difference between the two results is that the dimensionless system has better performance: such as lower overshot (by about 20%), and lower control effort (by about 25%). Perhaps another most important difference in the two systems is the stability margin. This can clearly be deducted from the fact that the dimensionless representation has more allowable perturbation (*margin* $\cong$ *1.27*) compared to the dimensional representation (*margin* $\cong$ *1.01*). The results are summarized in Table **5-2**.

Table **5-2**: Summary of the numerical results

| | Dimensional representation | Dimensionless representation | comparison |
|---|---|---|---|
| $H_\infty$ norm | 4.977 < 5.0 | 1.576 < 2.0 | Better margin 1.01 vs 1.27 |
| Overshoot | 2.2 | 1.8 | Improved overshoot by ~ 20% |
| Control action | 0.95 | 0.7 | Less control effort by ~ 25% |

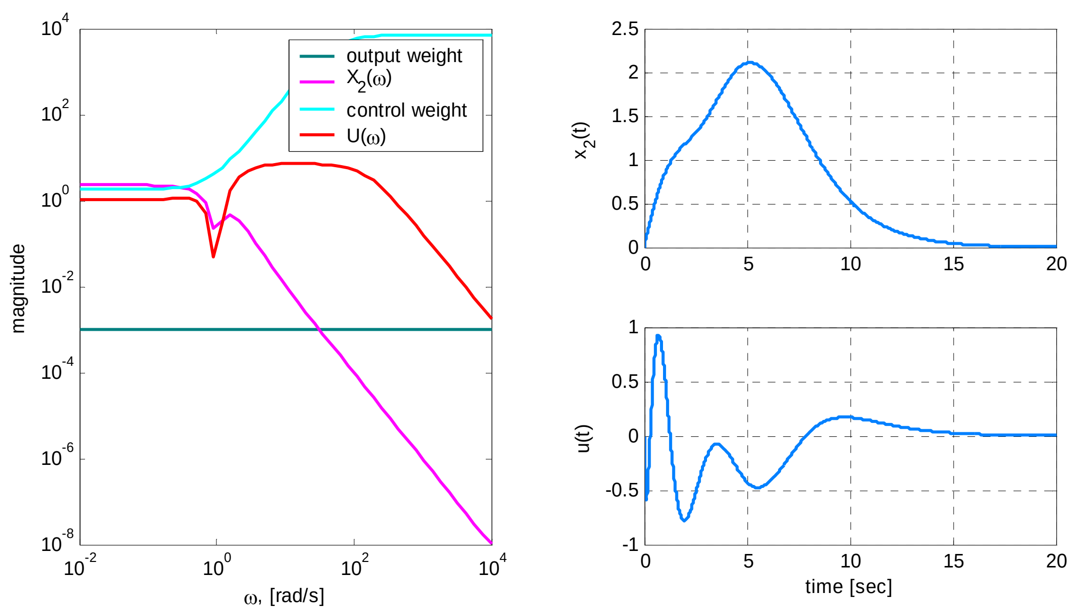



Figure **5-5**: dimensional system, $H_\infty$ norm 4.977 < 5.0.

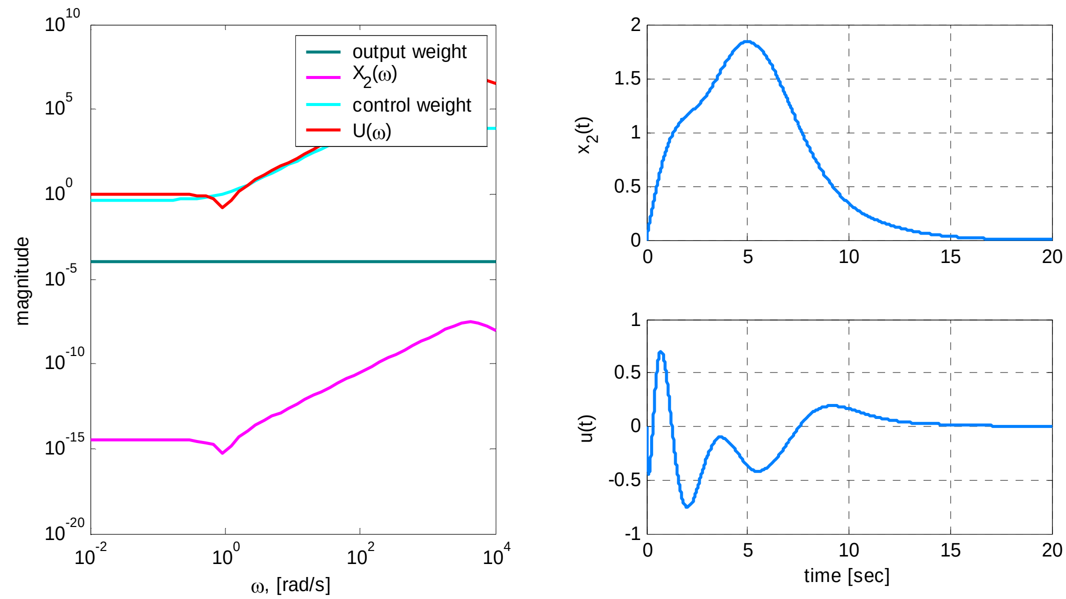



Figure **5-6**: dimensionless system, $H_\infty$ norm 1.576 < 2.0.

The concepts and comparison criteria discussed are general and require more discussion. The following section explains these concepts using a simpler problem, a mass-spring-damper, and focusing on robust stability.

## 5.5 Robust Stability Margin and Allowable Parametric Perturbation

In this section the allowable parametric perturbation in the two representations are discussed by comparison of the controller design in the dimensionless and the dimensional domains. For this purpose a simple mass-spring-damper system, Figure **5-7** , is used as an example. Assume that all the three parameters of the system, $m$ , $c$ and $k$ , exhibit some uncertainty and only the nominal values of these parameters ($m_0$, $c_0$, $k_0$) and the bound of their variations ($\delta_m^{\max}$, $\delta_c^{\max}$, $\delta_m^{\max}$) are known.

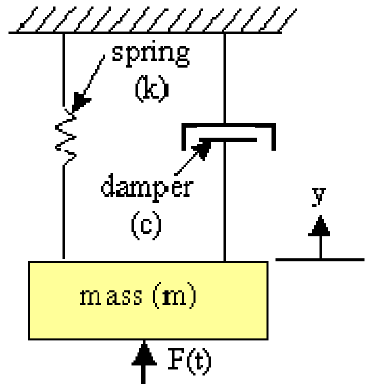



Figure **5-7**: A mass-spring-damper system

Therefore, the parameters can be expressed in terms of the nominal and uncertainty bounds, Eq. **5.10**. Note that $\delta_m$, $\delta_c$, and $\delta_k$ can have a negative or positive value.

$$\begin{aligned} m &= m_0 + \delta_m \\ c &= c_0 + \delta_c \\ k &= k_0 + \delta_k \end{aligned} \qquad \textbf{5.10}$$

with, $|\delta_m| \le \delta_m^{\max}$, $|\delta_c| \le \delta_c^{\max}$, $|\delta_k| \le \delta_k^{\max}$

The equation of motion (EOM) of the systems with these uncertain parameters is determined by substituting the expression for the parameters into the general EOM of the system as indicated by Eq. **5.11**.

$$\begin{aligned} & m\ddot{y} + c\dot{y} + ky = F \\ & \Rightarrow (m_0 + \delta_m)\ddot{y} + (c_0 + \delta_c)\dot{y} + (k_0 + \delta_k)y = F \end{aligned} \qquad \textbf{5.11}$$

with, $|\delta_m| \le \delta_m^{\max}$, $|\delta_c| \le \delta_c^{\max}$, $|\delta_k| \le \delta_k^{\max}$

The model is converted into block diagram, Figure **5-8** , in order to separate the uncertainty from the known nominal model and represent the system in LFT form.

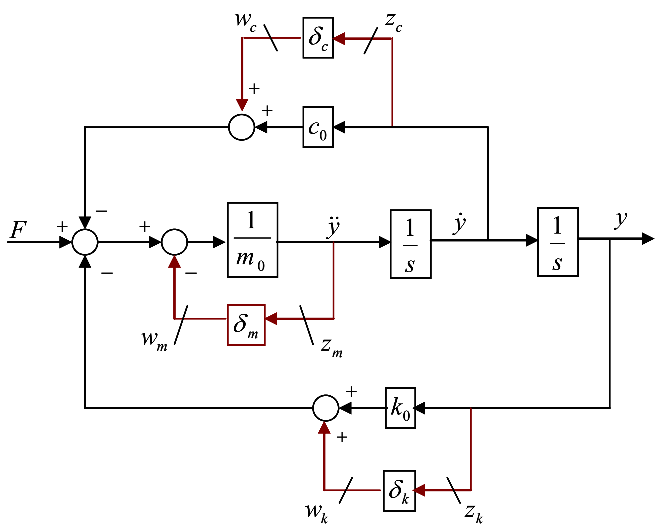



Figure **5-8**: Block diagram of the uncertain mass-spring-damper system: dimensional

The uncertainty input-output relationship is determined from the block diagram, Figure **5-8**, and given by Eq. **5.12** indicates that the uncertainty block is diagonal.

$$\begin{bmatrix} w_m \\ w_c \\ w_k \end{bmatrix} = \begin{bmatrix} \delta_m & 0 & 0 \\ 0 & \delta_c & 0 \\ 0 & 0 & \delta_k \end{bmatrix} \begin{bmatrix} z_m \\ z_c \\ z_k \end{bmatrix} \qquad \textbf{5.12}$$

The dimensionless model of the system is determined using $\pi = \frac{c}{\sqrt{mk}}$, $\bar{y} = y\frac{k}{mg}$, $\bar{F} = F\frac{1}{mg}$ and $\tau = t\sqrt{\frac{k}{m}}$ to be the form shown in Eq. **5.13**. It has only one parameter, $\pi$. The nominal value, $\pi_0$, and uncertainty, $\delta_\pi$, of the parameter $\pi$ is then solved from the nominal parameters, $m_0$, $c_0$, $k_0$, and uncertainties, $\delta_m^{\max}$, $\delta_c^{\max}$, $\delta_k^{\max}$, of the dimensional system given in Eq. **5.10**. The results are given in Eq. **5.15**.

$$\bar{y}'' + \pi \cdot \bar{y}' + \bar{y} = \bar{F}$$
$$\Rightarrow \bar{y}'' + (\pi_0 + \delta_\pi)\bar{y}' + \bar{y} = \bar{F} \qquad \textbf{5.13}$$

where, the derivative $(') \equiv \frac{d}{d\tau} = \sqrt{\frac{m}{k}}\frac{d}{dt}$

$$\pi_0 = \frac{c_0}{\sqrt{m_0 k_0}} \qquad \textbf{5.14}$$

and

$$\delta_\pi = \frac{c}{\sqrt{mk}} - \frac{c_0}{\sqrt{m_0 k_0}} = \frac{c_0 + \delta_c}{\sqrt{(m_0 + \delta_m)(k_0 + \delta_k)}} - \frac{c_0}{\sqrt{m_0 k_0}} \qquad \textbf{5.15}$$

The block diagram representation of this model, Figure **5-9** , has an uncertainty description of unity dimension, contrary to the 3x3 of the dimensional model description.

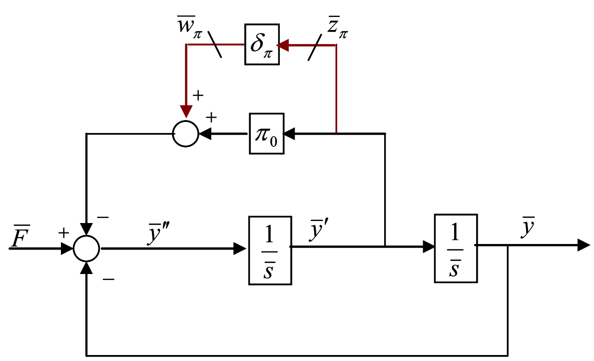



Figure **5-9**: Block diagram of the uncertain mass-spring-damper system: dimensionless

### 5.5.1 Return Difference Matrix Determinant Condition to Robust Stability

The robust stability condition is sometimes given in terms of the return difference matrix determinant condition, Eq. **5.16**, which is very important in the case of parametric or structured uncertainty [5, 26].

$$Robust\ stability \Leftrightarrow \det\left[I - M(j\omega)\Delta\right] \neq 0, \forall \Delta \in \mathbf{\Delta}, \forall \omega \qquad \mathbf{5.16}$$

where $M(j\omega)$ is the closed loop transfer matrix evaluated at the $j\omega$ - axis, and $\mathbf{\Delta}$ is the set of all possible uncertainties. For parametric uncertainty the uncertainty block, $\Delta$, is diagonal or block diagonal, and particularly for the mass-spring-damper system the uncertainty is strictly diagonal. Therefore, for the dimensional representation, the condition given in Eq. **5.16** can be reduced to the condition given in Eq. **5.17**. The maximum allowable perturbation (uncertainty) in this case is equal to the minimum values of the uncertainties, $\delta_m, \delta_c$, $\delta_k$, that violate the robust stability condition, Eq. **5.17**.

$$\begin{gathered} Robust\ stability \Leftrightarrow \det\left[I - M(j\omega)\Delta\right] \neq 0, \forall \Delta \in \mathbf{\Delta}, \forall \omega \\ \Leftrightarrow \det\left[1 - M(j\omega) \cdot diag\left[\delta_k \quad \delta_c \quad \delta_m\right]\right] \neq 0, \\ \forall \omega, \forall \delta_i \in \left[-\delta_i^{\max}, \delta_i^{\max}\right], i = \{m, c, k\} \end{gathered} \qquad \mathbf{5.17}$$

For the dimensionless representation, Eq. **5.13**, $M(j\omega)$ is a transfer function evaluated at the $j\omega$ - axis. The robust stability condition for this case is similarly given by Eq. **5.18** and the maximum allowable uncertainty is equal to the minimum value of the uncertainty $\delta_\pi$ that violates the robust stability condition Eq. **5.18**.

$$\begin{gathered} Robust\ stability \Leftrightarrow \det\left[I - M(j\omega)\Delta\right] \neq 0, \forall \Delta \in \mathbf{\Delta}, \forall \omega \\ \Leftrightarrow \left(1 - M(j\omega)\delta_\pi\right) \neq 0, \quad \forall \delta_\pi \in \left[-\delta_\pi^{\max}, \delta_\pi^{\max}\right], \forall \omega \end{gathered} \qquad \mathbf{5.18}$$

For a comparative study of the two representations a robust controller is designed for both the dimensional and dimensionless representation using MATLAB (see Appendix **A.3**). The parameter values $c_o = m_o = k_o = 1.0$ are used in this numerical example. A robust control analysis is performed on both closed loop systems to study the range of the allowable perturbation. Applying the stability conditions of Eq. **5.17** by evaluating it at the frequency that gives maximum singular value, $\omega_{\bar{\sigma}}$, yields a complex frequency point response of the closed loop transfer matrix of the dimensional model: $M(j\omega_{\bar{\sigma}})$. For this particular numerical example, the maximum singular value is determined as $\bar{\sigma}\left(M(s)\right)=1.733$ and the frequency at which this occurs is $\omega_{\bar{\sigma}}=395.8\,rad/s$. To determine for the values of the uncertainties, $\delta_m, \delta_c$, $\delta_k$, that violate robust stability one need to determine the values of these uncertainties that satisfy Eq. **5.19**.

$$\det\left[1-M(j\omega_{\bar{\sigma}})\cdot diag\left[\delta_k \quad \delta_c \quad \delta_m\right]\right]=0, \qquad \textbf{5.19}$$

This yields a complex relationship of the different uncertainties and equating the real and complex parts and solving yields Eq. **5.20**. Similarly, evaluating $M(j\omega_{\bar{\sigma}})$ in Eq. **5.18** for the closed-loop dimensionless representations relation among the uncertainty parameters as defined by Eq. **5.20** and Eq. **5.21**.

$$\begin{aligned} \delta_k &= \frac{\left(1+4.49564\delta_c\right)}{6377.234} \\ \delta_m &= 0.1+0.89921\delta_c \end{aligned} \qquad \textbf{5.20}$$

$$\delta_\pi = 0.5771 \qquad \textbf{5.21}$$

From Eq. **5.15**, substituting for $\delta_\pi$, we have Eq. **5.22**

$$\delta_\pi = \frac{1.0+\delta_c}{\sqrt{(1.0+\delta_m)(1.0+\delta_k)}} - 1.0 = 0.5771$$

$$\delta_c = 1.5771\sqrt{(1.0+\delta_m)(1.0+\delta_k)} - 1.0 \tag{5.22}$$

The combination of parametric perturbations that violate the robust stability condition is shown in Figure **5-10**. The green surface represents the location where, if a combination of the uncertain parameters lies on this surface, then the dimensionless closed-loop system will be unstable. Similarly, when a combination of the uncertain parameters lies on the red line then the dimensional closed loop system will be unstable.

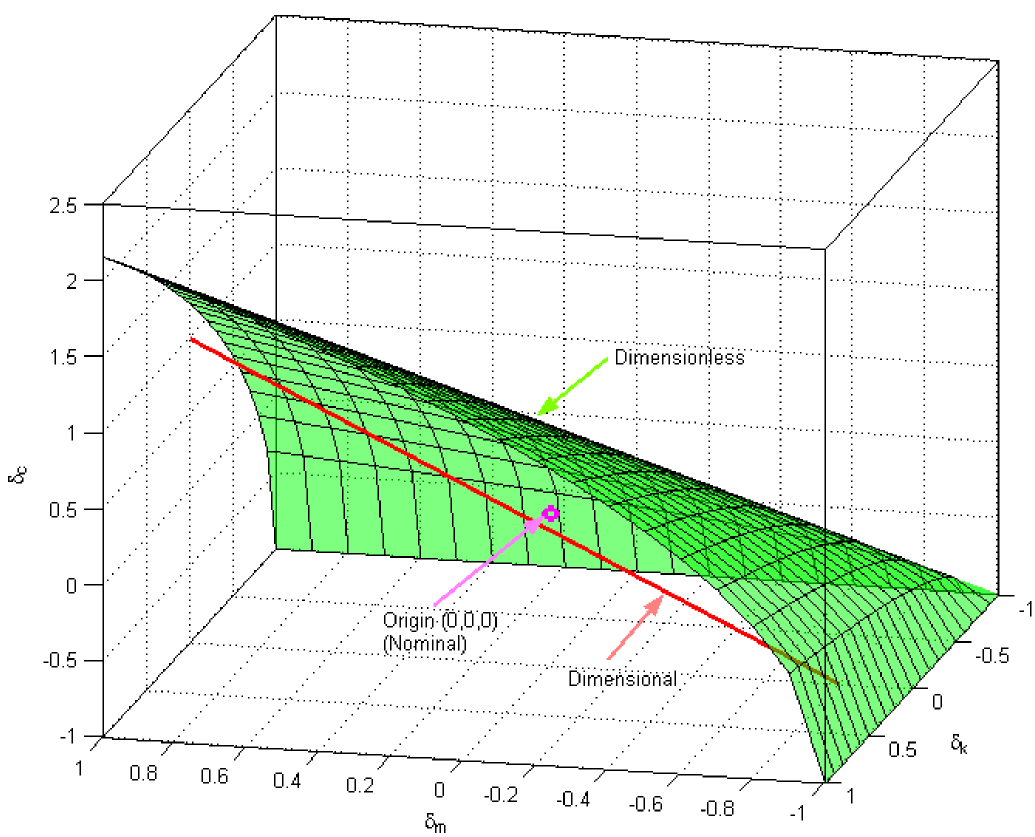



Figure **5-10**: Parametric perturbation space

Robust control is based on the worst case scenario, so consequently the comparison of the two representations is performed in this respect. The origin of the parametric perturbation space, Figure **5-10**, represents the nominal plant. The point on the instability line/surface that yields the *minimum* distance to the origin (nominal point) is the worst-case scenario, e.g. the maximum allowable perturbation based on maintaining stability. For the current example, a plot of the distance function between the origin and every vertex in the parametric perturbation space is shown in Figure **5-11**.

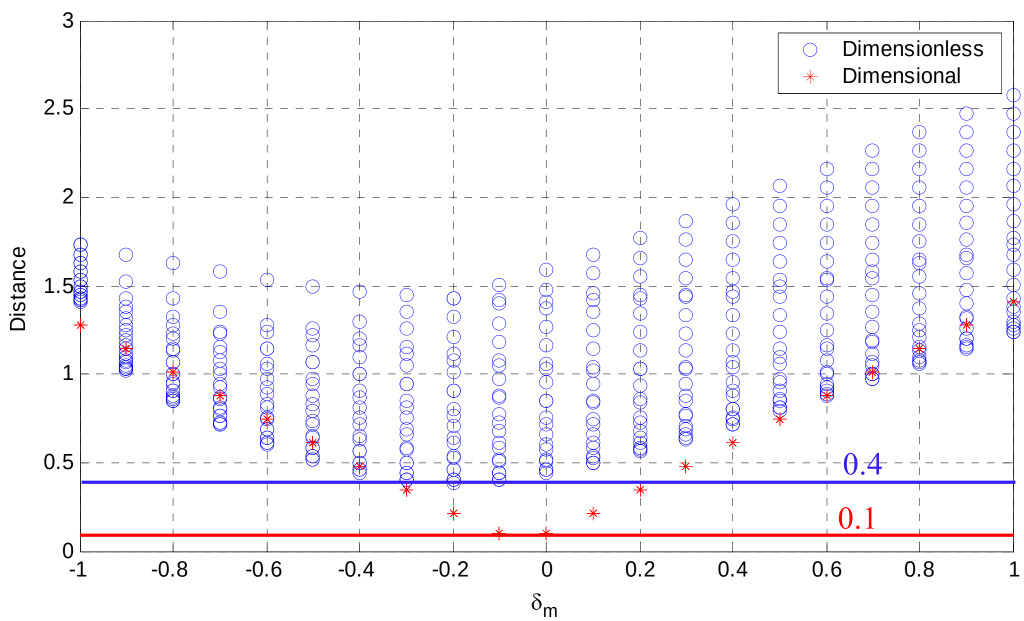



Figure **5-11**: The distance from nominal to instability line/surface

The horizontal axis is $\delta_m$ and $\delta_k$ is directed into the paper and $\delta_c$ is dependent on both $\delta_m$ and $\delta_k$ and is not plotted. The plot for the dimensionless uncertainty has many points because each point represents the distance from the origin to a vertex on the surface, while for the dimensional it represents the distance to a point on the line, hence fewer points. Based on this result, one can make the assertion that dimensional system has lower allowable parametric perturbation than the dimensionless approach. The values for the dimensional and dimensionless are 0.1 and 0.4, respectively.

## 5.6 Some Practical Limitations of the Approach

The presented method of parametric uncertainty block size reduction has three potential limitations. These limitations are problem dependent and therefore their drawbacks vary from problem to problem. The details of these potential problems are discussed and possible solutions are recommended.

### 5.6.1 Uncertainty Interval Expansion

The first drawback is that the dimensionless uncertainty description may have wider uncertainty bound than the dimensional case. For example, in the ACC benchmark problem considered above, the individual parametric uncertainty are bounded as $|\delta_i| \leq 0.2$. However, after the dimensional transformation, the dimensionless representation has one parametric uncertainty that must be bounded as $|\delta_{\pi 1}| \leq 0.5$ to capture the equivalent parametric variation from the original, dimensioned representation. Therefore there is a tradeoff between reducing the number of uncertainty parameters and widening of the uncertainty bounds.

While this interval widening may play a negative role in trying to reduce the conservativeness of the problem, for some problems, the merits outweigh the drawbacks. As an example would be the numerical example above. Further, for most engineering systems – the vehicle example in the next chapter, for instance – the uncertainties in one parameter (mass for example) will be likely couple with uncertainty in another (inertia) such that the actual uncertainty in a $\pi$-parameter is much less than the uncertainty that would be calculated from extremes in the dimensioned parameters. But there is no guarantee that this will always be the case and additional investigation or side-by-side comparison is recommended for each system.

### 5.6.2 Input/Output Uncertainty Scaling

The second limitation is in regard to the implementation of robust controller designed in the dimensionless domain. A dimensionless controller cannot be directly implemented because physical systems are dimensioned. In other words, the dimensional signal output from the real plant has to be transformed into a dimensionless signal to be used as feedback into the dimensionless controller. The output of the dimensionless controller then needs to be transformed back into a dimensioned control signal since it is used to actuate the plant. In both transformations, the signals may be scaled by uncertain parameters. The scaling may results in input/output uncertainty on the system. There are two scenarios: 1) The input to the controller (sensor signal) and the output share the same type physical quantity (such as Force-Force, displacement-displacement, torque-work, etc…), any uncertainty on the input-output scaling of the controller cancels each other and there would be no difficulty in the implementation of the controller. 2) The input-output of the controller are not the same type physical quantities (such as displacement-force, displacement-pressure, etc…) then there may be an uncertainty to the input-output scaling that would therefore make the output of the controller uncertain.

The issue with uncertain scaling, the second scenario discussed above, may be mitigated by re-inclusion of some of the uncertainty back into the design model as a gain uncertainty. The re-inclusion of this uncertainty gain at the input/output of the dimensionless controller is straightforward, but at the cost of the overall reduction of the uncertain parameters. This is illustrated using the following example.

In the ACC benchmark example considered above, the controller has displacement as an input and force as an output as shown in Figure **5-12** below. Referring to the scaling factors of the two signals, the dimensions obviously do not cancel, i.e. they are related by the stiffness which is an uncertain parameter.

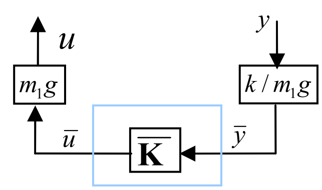



Figure **5-12**: Input-output scaling of the controller

In this case, we introduce an additional constant stiffness parameter instead and then the original stiffness parameter will me moved to $B_D$ and will have one more dimensionless uncertain parameter. The newly introduced constant stiffness variable will replace the input-output transformation in Figure **5-12** and the original stiffness parameter will now re-cast into the plant. The new plant will have the uncertainty description of the form given by Eq. **5.23** contrary to Eq. **5.9**. In this case, the method reduces the parametric uncertainty by one instead of by two. Note that $\delta_{\pi 2}$ is due to the uncertain stiffness parameter that is re-cast back to the plant.

$$\begin{bmatrix} \overline{w}_{\pi 1} \\ \overline{w}_{\pi 2} \\ \overline{w} \end{bmatrix} = \begin{bmatrix} \delta_{\pi 1} & 0 & 0 & 0 \\ 0 & \delta_{\pi 3} & 0 & 0 \\ 0 & 0 & \delta_y & \delta_u \end{bmatrix} \begin{bmatrix} z_{\pi 1} \\ z_{\pi 3} \\ \overline{z}_{\overline{y}} \\ \overline{z}_{\overline{u}} \end{bmatrix} \qquad \textbf{5.23}$$

### 5.6.3 Uncertainty in Time Scaling

This arises from the scaling of time by uncertain parameters whose only nominal values are known. For the mass-spring-damper systems the time scaling is $\sqrt{k/m}$ and both $k$ and $m$ are assumed to be uncertain. This can potentially deteriorate the robustness of the closed-loop system. In this case the dimensionless robust control system can be transformed to the dimensional domain. By closing the loop with the dimensional plant, a robust analysis can be performed.

For example, for the above mass-spring-damper example, the controller is transformed to the dimensional domain and $\mu$-analysis is performed on the closed-loop system. The transformed dimensionless controller still shows better stability margin than its dimensional counterpart. The structured singular value for dimensional controller is 1.0 (gain margin) while for the dimensionless it is 0.6645. Further, details as well as MATLAB code can be found in Appendix **A.3**.

## 5.7 Summary

A method known as dimensional transformation was presented that can be used to reduce the number of parametric uncertainty in some robust control problems. This method was demonstrated on the 1990 benchmark problem.

It was shown in the general structured uncertainty case that a lower $H_\infty$ norm is achieved by using dimensional transformation to reduce the size of the uncertainty block in the system description. This lower norm should increase the allowable perturbation of the system in the new representation. Also, a better performance in the form of reduced overshoot and control effort was achieved using the dimensionless representation compared to the dimensional representation.

A study of the two representations based on the return difference matrix determinant condition of robust stability has shown more clearly that the dimensionless representation gives a higher allowable parametric perturbation than the dimensional representation by a factor of about four.

Finally some practical limitations of the method were presented. These limitations are highly problem dependent, but are generally mitigated by re-inclusion of uncertainty into the system representation to account for uncertain scaling factors. For the uncertainty time scaling case, a thorough check is recommended to confirm the robustness of the controller under uncertain model or signal scaling.

## 5.8 References


[1] A. Packard and K. Zhou, "Improved Upper Bounds for the Structured Singular Value," Proceedings of the 28th IEEE Conference on Decision and Control, Tampa, Florida, 1989.

[2] M. K. H. Fan, A. L. Tits, and J. C. Doyle, "Robustness in the Presence of Mixed Parametric Uncertainty and Unmodeled Dynamics," *IEEE Transaction on Automatic Control*, vol. 36, pp. 25-38, 1991.

[3] G. M. Voorsluijs and J. A. Mulder, "Parametric-Dependent Robust Control for a Rotorcraft UAV," AIAA Guidance, Navigation, and Control Conference and Exhibit, San Francisco, CA, USA, 2005.

[4] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control, Design and Analysis*. Chichester, New York: John Wiley & Sons Ltd., 1996.

[5] R. Sánchez-Peña and M. Sznaier, *Robust Systems, Theory and Applications*. New York, NY: John Wiley & Sons Ltd., 1998.

[6] G. Zames, "Feedback and Optimal Sensitivity: Model Reference Transformations, Multiplicative Seminorms, and Approximate Inverse," *IEEE Transaction on Automatic Control*, vol. 26, pp. 301-320, 1981.

[7] C. M. Belcastro, "Parametric Uncertainty Modeling: An Overview," Proceedings of the 1998 American Control Conference, Philadelphia, Pennyslvania, 1998.

[8] M. Fu, "Parametric Approach to Robust Stability with both Parametric and Nonparametric Uncertainties," Proceedings of the 31st IEEE Conference on Decision and Control, Tucson, Arizona, 1992.

[9] A. Sideris, "Robust Performance in Systems with Parametric Uncertainty," Proceedings of the 28th IEEE Conference on Decision and Control, Tampa, Florida, 1989.

[10] J. M. Collado and J. A. Rojas, "New Technique for Robust Stability," Proceedings of the 31st IEEE Conference on Decision and Control, Tucson, Arizona, 1992.

[11] J. C. Doyle, "Analysis of Feedback Systems with Structured Uncertainties," Proceedings of IEE, Part-D, 1982.

[12] M. G. Safonov, "Stability Margins of Diagonally Perturbed Multivariable Feedback Systems," IEE Proceedings, 1982.

[13] J. C. Doyle, "Structured Uncertainty in Control System Design," Proceeding of 24th IEEE Conference on Decision and Control, 1985.

[14] J. L. Lin, I. Postlethwaite, and D. W. Wu, "μ-K iteration: A new algorithm for μ-synthesis," *Automatica*, vol. 29, pp. 219-224, 1993.

[15] V. R. Basker, H. A. Latchman, H. M. Mahon, and O. D. Crisalle, "A New Pespective on Computing Robust Stability Margins for Complex Parametric Uncertainties," Proceedings of the 1998 American Control Conference, Philadelphia, Pennyslvania, 1998.

[16] A. Sideris and R. S. Sanchez Pena, "Robustness Margin Calculation with Dynamic and Real Parametric Uncertainty," *IEEE Transaction on Automatic Control*, vol. 35, pp. 970-974, 1990.

[17] A. Packard, M. K. H. Fan, and J. C. Doyle, "A Power Method for the Structured Singular Value," Proceedings of the 27th IEEE Conference on Decision and Control, Austin, Texas, 1988.

[18] H. Ito, "An Approach to Robust Control of Sampled-Data Systems Having Finite-Dimensional LTI and Parametric Uncertainties," Proceedings of the 35th IEEE Conference on Decision and Control, Kobe, Japan, 1996.

[19] P. Shi, L. Xie, and C. E. de Souza, "Output feedback control of sampled-data systems with parametric uncertainties," Proceedings of the 31st IEEE Conference on Decision and Control, Tucson, Arizona, 1992.

[20] L. H. Keel and S. P. Bhattacharyya, "Robust, Fragile, or Optimal," *IEEE Transaction on Automatic Control*, vol. 42, pp. 1098-1105, 1997.

[21] R. H. C. Takahashi, D. A. Durtra, R. M. Palhares, and P. L. D. Peres, "On Robust Non-Fragile Static State-Feedback Controller Synthesis," Proceedings of the 39th IEEE Confererence on Decision and Control, Sydney, Australia, 2000.

[22] D. Famularo, C. Abdallah, A. Jadbabaie, P. Dorato, and W. M. Haddad, "Robust Non-fragile LQ Controllers: The Static State Feedback Case," Proceedings of the 1998 American Control Conference, Philadelphia, Pennsylvania, 1998.

[23] W. M. Haddad and J. H. Corrado, "Robust Resilient Dynamic Controllers for Systems with Parametric Uncertainty and Controller Gain Variations," Proceedings of the 1998 American Control Conference, Philadelphia, 1998.

[24] P. M. Makila, "Comments on "Robust, Fragile, or Optimal," *IEEE Transaction on Automatic Control*, vol. 43, pp. 1265-1268, 1998.

[25] B. Wie and D. S. Bernstein, "A Benchmark Problem for Robust Controller Design," Proceedings of American Control Conference, San Diego, CA, 1990.

[26] D. Bates and Postlethwaite, *Robust Multivariable Control of Aerospace Systems*, vol. 03. Delft: Delft University Press, 2002.

# Chapter 6

# Vehicle Autopilot: A Simultaneous Robust Control through Parametric Adaptation

## 6.1 Introduction

This chapter discusses a method for robust simultaneous control technique for systems whose system parameters are inherently coupled. Assuming the existence of inherent interrelationships between the different parameters, the current approach exploits this property for a better robust control design. The chapter shows the potential of the proposed method for designing a unified robust controller that can be implemented through parametric adaptation. This chapter also establishes such a controller is robust, adaptive and modular.

Human- or naturally-optimized systems will likely exhibit a property where all parameters of the system are each interrelated. This arises because the key dynamic parameters of a system are generally the same that must be optimized to satisfy design criteria on the system behavior. This will naturally couple the key system parameters in predictable relationships. One such example is the collective dynamic behavior of passenger vehicles. For example: a large vehicle tends to be long, is heavier and has a larger mass moment of inertia. Additional generalizations can be made between vehicle size and the tire’s force generation performance, the suspension behavior, etc. These relationships between length, mass, inertia, etc. obviously do not follow an exact functional relationship. But if one simply knows that the system under consideration is a modern production vehicle, one can infer general estimates of all parameters often given just one parameter, mass for instance. This inference can be generalized to functional parameter relationships, and such relationships should be expected for any optimized system.

This concept of parametrically constrained engineering systems can be best explained with the help of Figure **6-1**. Consider three systems that are parametrically different and as indicated in the $p_1 - p_2 - p_3$ space. These systems are represented by $G_1, G_2, G_3$ and all enclosed inside the volume $S$, Figure **6-1**(*a*).

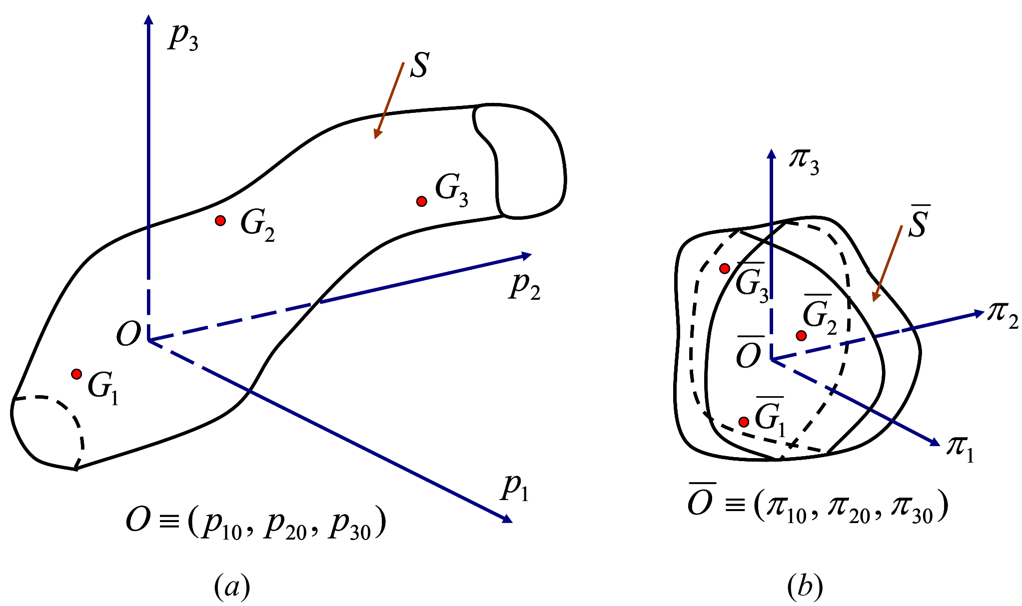



Figure **6-1**: Schematic of systems in parametric space

In the case of passenger vehicles, $G_1$ may represent a compact car, $G_2$ mid-size and $G_3$ a luxury size sedan. One can attempt to design a single robust controller to simultaneously stabilize the three plants. However, as the number of plants increases in a way that extends the solid $S$ further, it may be very difficult if not impossible to synthesize a single controller to stabilize all the plants. Further one has to consider other plants represented by parametric variation within a sphere that encloses $S$. Many of these parameter combinations can never physically occur. Therefore controllers that consider these plants as representative of key constraints on system performance or robustness may result in a highly conservative controller when implemented on actual systems. Hence, when collective aggregates of dissimilarly-sized systems are considered, it becomes very difficult to find a controller that satisfies all robustness and performance conditions for all systems.

The technique described in this thesis addresses this problem by transforming the system representations to dimensionless domain using the method of dimensional transformation in order to convert the variation of the plants as shown in Figure **6-1**(*b*). This is because the transformation to the dimensionless representation will be selected such that it transforms only the solid $S$, not the sphere that encloses $S$.

The remaining part of the chapter is organized as follows: First review of relevant literature is presented in section **4.2**. The general framework of the proposed technique is discussed in section **6.3**. The dynamic model of the system used to demonstrate the technique is also discussed in this section. A mathematical definition of dimensional transformation along with some examples as well as the distribution of vehicle parameters collected from literature is also included. Section **6.4** focuses on the process of developing the current technique and how to setup the general problem as well as the solution. The different nominal and bounds of the parameters are defined in this section and numerical results are also presented. Section **6.5** presents the experimental testing of the technique using the scaled vehicle on a rolling-roadway simulator system. Finally, summary of the main points and results is given in section **6.6**.

## 6.2 Literature Review

In the early days of robust control research, most control engineers focused on the theoretical developments of the robust control technique focusing on the development of efficient mathematical solutions and algorithms for the control synthesis. However, due to the rapid development of computational tools, i.e., faster computers and processors, the robust control techniques have been gaining more applications as practical engineering tools rather than mathematical concepts [1, 2]. The literature of robust control application/implementation is very extensive and only a few examples are discussed here specifically focused on the aerospace and automotive industries. In the aerospace industry, a flight test of an $H_\infty$ control law demonstrated by a scheduled $H_\infty$ loop-shaping flight control was first reported in [3]. A first-ever test flight of an LPV flight control law designed using the method of $H_\infty$ loop-shaping was reported in [4]. The application of $\mu$-synthesis techniques were also reported in [5] for flexible structure and [6, 7] for missile autopilots. Additional implementation and applications are reported in [8, 9] for F-14 military aircraft, [10, 11] for civil transport aircraft, [12] for UAVs and others in [13-15].

The application/implementation of the robust control technique in the automotive industries is not as extensive as in the aerospace industry as reported in public literature. However, robust control implementation are gaining increased interest in applications of

Automated Highway Systems (AHS) [16, 17]. A robust $H_\infty$ loop-shaping controller was designed in [16], and in [17] a nonlinear robust controller was developed for lateral control of heavy trucks in automated highways. In most vehicle models, the vehicle velocity appears as a free parameter. Gain-scheduling is required in this case due to the significant changes in the vehicle dynamic model as a function of velocity, changes that sometimes change an open-loop stable vehicle model to an unstable model with increasing speeds. To address this velocity dependence, a gain-scheduling controller was designed in [18] and an LPV controller in [19]. Additional applications are described in [20-24].

While scaling theory is an old subject and has been applied to dynamical and structural systems, its application to controls has been started only during the last decade, to the best of the author's knowledge. One of the most recent and well developed work in this area is the works of Brennan and Alleyne [22, 25, 26]. Previous work by Brennan [22] has shown the advantages of using the dimensionless representation in vehicles for robust control design. Specially, Brennan [22] has shown the achievement of tight frequency domain variations using dimensionless vehicle models. The tight frequency domain distribution allows for small plant variations from the nominal model finally resulting in smaller uncertainty bounds.

The current work is very different from the previous works in three most important ways. First, the previous works used a general stacked sensitivity approach which results in a dynamic uncertainty model. The current work models system-to-system variations as a parametric uncertainty which, unlike to the dynamic uncertainty, is less conservative. Secondly, the current work uses the general $H_\infty$ - synthesis and the $\mu$ - synthesis/analysis to better account for structure in the uncertainty model, which haven't been done in the previous work.

## 6.3 Framework of the Technique

Some concepts and systems that help present the materials are defined in this section. The subsections are organized as follows: a vehicle model used to demonstrate the current technique, a model known as the bicycle model, is first presented. Following, the dimensional transformation method is presented and the operator of the transformation on different quantities as well as systems is discussed in detail. Finally, a general setup of the proposed technique is discussed.

### 6.3.1 The Bicycle Model

There are several vehicle models with different level of complexities and different application. The choice of the vehicle model depends on the type of dynamics that are of interest. In some cases one type of dynamics can be represented by different models having different level of complexity and level of details. The interest to this work is on lateral control of the vehicle using a steering control input. Apart from actuator dynamics, the two dominant motions of the automatic vehicle steering control are yaw and lateral motions.

A two degree-of-freedom (DOF) planar vehicle dynamic model commonly used to describe these motions is called the bicycle model [22, 27]. In this model, the coupling between the roll and lateral modes is not considered. The dynamic model is herein expressed in road fixed error coordinates, Figure **6-2**. The general bicycle model may have more inputs, however only front steering input is considered here. The equation of motion (EOM) of this 2-DOF model is given by Eq. **6.1**.

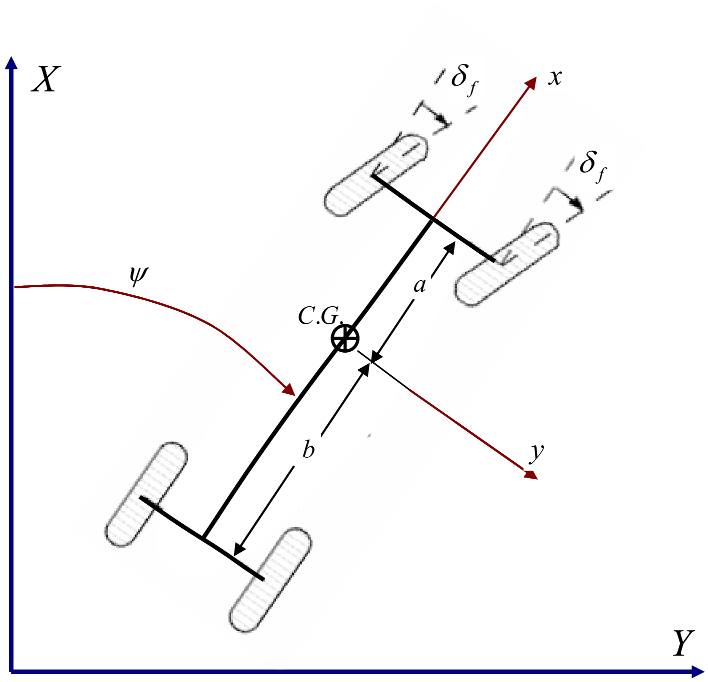



Figure **6-2**: Schematics of the bicycle model. X-Y and x-y are road (path) and body fixed coordinates, respectively.

$$
\begin{aligned}
m \cdot \ddot{y} &= -\frac{C_{\alpha f} + C_{\alpha r}}{U} \cdot \dot{y} + \left(C_{\alpha f} + C_{\alpha r}\right) \cdot \psi - \frac{a \cdot C_{\alpha f} - b \cdot C_{\alpha r}}{U} \cdot \dot{\psi} + C_{\alpha f} \cdot \delta_f \\
I_z \cdot \ddot{\psi} &= -\frac{a \cdot C_{\alpha f} - b \cdot C_{\alpha r}}{U} \cdot \dot{y} + \left(a \cdot C_{\alpha f} - b \cdot C_{\alpha r}\right) \cdot \psi - \frac{a^2 \cdot C_{\alpha f} + b^2 \cdot C_{\alpha r}}{U} \cdot \dot{\psi} + a \cdot C_{\alpha f} \cdot \delta_f
\end{aligned} \qquad \textbf{6.1}
$$

where, the different positions, speeds and input are defined as:

$y$ : lateral position,

$\dot{y}$ : lateral velocity,

$\psi$ : yaw angle,

$\dot{\psi}$ : yaw rate, and

$\delta_f$ : front steering input.

and the vehicle parameters are defined as:

$m$ : vehicle mass,

$I_z$ : vehicle moment of inertia,

$U$ : vehicle longitudinal velocity,

$a$ : distance between the center of gravity (C.G.) and the front axle,

$b$ : distance between the center of gravity (C.G.) and the rear axle,

$L$ : vehicle length between the front and rear axels ($= a + b$),

$C_{\alpha f}$ : cornering stiffness of the front 2 tires, and

$C_{\alpha r}$ : cornering stiffness of the rear 2 tires.

Cornering stiffness is the ratio of the lateral force to the slip angle, assuming approximately linear relationship between the two. The cornering stiffness is dependent on the tire-road interaction and generally determined experimentally. The interested reader is referred to [28, 29] for a detailed discussion about the cornering stiffness.

The bicycle model can be represented in state-space form [22], Eq. **6.2**, by choosing the state vector, $\mathbf{x} \equiv [y \quad \dot{y} \quad \psi \quad \dot{\psi}]^T$ and the control input, $u \equiv \delta_f$.

$$\dot{\mathbf{x}} = A\mathbf{x} + Bu$$
$$\mathbf{y} = C\mathbf{x} + Du \qquad \mathbf{6.2}$$

where, the system matrices are given by Eq. **6.3**.

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{f_1}{mU} & \frac{f_1}{m} & -\frac{f_2}{mU} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{f_2}{I_z U} & \frac{f_2}{I_z} & -\frac{f_3}{I_z U} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \frac{C_{\alpha f}}{m} \\ 0 \\ \frac{a \cdot C_{\alpha f}}{I_z} \end{bmatrix} \qquad \mathbf{6.3}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad D = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

with, $f_1$, $f_2$ and $f_3$ as defined in Eq. **6.4**.

$$f_1 = C_{\alpha f} + C_{\alpha r}$$
$$f_2 = a \cdot C_{\alpha f} - b \cdot C_{\alpha r} \qquad \mathbf{6.4}$$
$$f_3 = a^2 \cdot C_{\alpha f} + b^2 \cdot C_{\alpha r}$$

In this system, $a$, $b$, $L$, $m$, and $I_z$ increase with the increase of the size of the vehicle, therefore, they exhibit an inherent proportionality relationship. Additionally, the front and rear cornering stiffness increase/decrease with mass, and they generally change together in unison due to uniform road and similar tire conditions.

### 6.3.2 The Dimensional Transformation of Systems, $\Im_{\mathbf{D}}$

The dimensional transformation method discussed in Chapter 3 focuses on variable transformation and detailed steps were presented. In this chapter, however, the dimensional transformation for a system (dynamic model) is presented as a state- and time- normalization. For this purpose the transformation of variable is presented more formally. Before, defining the technique, some review of concepts and definitions from Chapter 3 are given.

### 6.3.2.1 Some Preliminaries

*The dimensional extraction operator,* $d_{v,e} = D(e,v)$, is an operator that extracts the units of the variable $v$ relative to the unit system $e$, a column vector by convention and results in a dimensional unit vector. This dimensional unit vector is nothing but the exponents of each unit systems used to describe the physical quantity $v$. To uniquely define this vector, one must specify both the unit system as well as the variable, $v$. For instance, the gravitational constant, $g = 9.81\ \mathrm{m/s^2}$, has dimensional units that can be represented in many unit systems. For the unit system, $e = [length \quad mass \quad time]^T$ the extraction operator yields $d_{v,e} = D(e,g) = [1 \quad 0 \quad -2]^T$, and in another unit system $e = [mass \quad force]^T$, the result of the operator is $d_{v,e} = D(e,g) = [-1 \quad 1]^T$.

*The Scaling Matrix,* $A_D$: is formed by the dimensional unit vectors of the variables (parameters) that are chosen as scaling parameters, also traditionally called repeating parameters. $A_D$ is a square matrix with its rows (and column) size equal to the row size of $e$ and must be a full rank. For example, for the bicycle model discussed before, choosing the unit system $e = [length \quad mass \quad time]^T$ and the scaling parameters

$\{m, L, U\}$, the scaling matrix $A_D$ is given by Eq. **6.5**, where the unit system $e$ remains the same for all the parameters and the operator is acted upon each parameter under the same unit system.

$$A_D = \begin{bmatrix} d_{M,e} & d_{U,e} & d_{L,e} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} \qquad \textbf{6.5}$$

The matrix $A_D$ is defined as a scaling matrix only if the dimensional unit vectors of all the scaling parameters (variables) are extracted with respect to a single unit system $e$.

### 6.3.2.2 Dimensional Transformation of a Variable, $\Gamma_D$

In this section the transformation of variables (parameters and signals) is discussed. From chapter 3, the vector solution of the transformation from dimensional to dimensionless variables is given by Eq. **6.6**. This solution represents the exponents of the scaling variables (refer to chapter 3 for details).

$$C_S = \left(-A_D^{-1} B_D\right)^T \qquad \textbf{6.6}$$

The matrix $B_D$ is formed by the dimensional unit vector of the remaining variables in the system that are to be scaled, also traditionally called non-repeating variables. For the bicycle model these variables include the parameters and the signals and, therefore, the matrix $B_D$ is given in Table **6-1**.

Table **6-1**: The dimensional unit vectors of the parameters and signals of the bicycle model

$$\begin{array}{c|ccccccccc|ccc|}
 & a & b & C_{\alpha f} & C_{\alpha r} & I_z & \delta_f & \psi & y & t & M & L & U \\
\hline
m & 1 & 1 & 1 & 1 & 2 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\
kg & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
s & 0 & 0 & -2 & -2 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 \\
\hline
 & \multicolumn{9}{c|}{\mathbf{B_D}} & \multicolumn{3}{c|}{\mathbf{A_D}}
\end{array}$$

Expanding the expression in Eq. **6.6** further and expressing $B_D$ in terms of its column vectors, as $B_D = \begin{bmatrix} d_{v_1,e} & d_{v_2,e} & \cdots & d_{v_n,e} \end{bmatrix}$, yield the expression for the vector solution $C_S$ Eq. **6.7**. This relation is used to define the dimensional transformation of individual variables.

$$C_S = -B_D^T A_D^{-T} = -\begin{bmatrix} d_{v_1,e}^T \\ d_{v_2,e}^T \\ \vdots \\ d_{v_n,e}^T \end{bmatrix} A_D^{-T} = \begin{bmatrix} -d_{v_1,e}^T \cdot A_D^{-T} \\ -d_{v_2,e}^T \cdot A_D^{-T} \\ \vdots \\ -d_{v_n,e}^T \cdot A_D^{-T} \end{bmatrix} \qquad \textbf{6.7}$$

*Definition 6.1: The dimensional transformation of a variable* $v$, $\Gamma_D(v,e,w)$: Given a variable $v$, unit systems $e$ and the scaling variables $\begin{bmatrix} w_1 & w_2 & \cdots & w_n \end{bmatrix}$, the dimensional transformation of $v$ to its corresponding dimensionless quantity is defined in Eq. **6.8**.

$$\bar{v} = \mathbf{\Gamma_D}(v,e,w) = v \cdot \prod_{i=1}^{n} w_i^{r_v(i)}, \text{ with, } r_v = -d_{v,e}^T \cdot A_D^{-T} \qquad \textbf{6.8}$$

For example, the parameters and signals of the bicycle model in Table **6-1** are transformed into their corresponding dimensionless variables using *definition 6.1*, Eq. **6.8**. In order to discuss the examples, the matrix $A_D^{-T}$, is numerically solved and is given by Eq. **6.9**.

$$A_D^{-T} = \left(A_D^{-1}\right)^T = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & -1 \end{bmatrix} \qquad \textbf{6.9}$$

The, following are some examples from the bicycle model parameters and signals of Table **6-1**, to illustrate the dimensional transformation of variables. Also for this example $w = \begin{bmatrix} m & L & U \end{bmatrix}$ and $e = \begin{bmatrix} length & mass & time \end{bmatrix}^T$.

Example 1: The distance between the C.G. and the front axle, $a$:

$$r_a = -d_{a,e}^T \cdot A_D^{-T} = -\begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & -1 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \end{bmatrix}$$

hence,

$$\begin{aligned} \pi_1 = \bar{a} = \mathbf{\Gamma_D}(a, e, w) &= a \cdot \prod_{i=1}^{3} w_i^{r_a(i)} \\ &= a \cdot \left(m^{r_a(1)} \cdot L^{r_a(2)} \cdot U^{r_a(3)}\right) \\ &= a \cdot \left(m^0 \cdot L^{-1} \cdot U^0\right) = a \cdot \frac{1}{L} \end{aligned}$$

Example 2: The cornering stiffness, $C_{\alpha f}$:

$$r_{C_{\alpha f}} = -d_{C_{\alpha f},e}^T \cdot A_D^{-T} = -\begin{bmatrix} 1 & 1 & -2 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & -1 \end{bmatrix} = \begin{bmatrix} -1 & 1 & -2 \end{bmatrix}$$

hence,

$$\begin{aligned}\pi_3 = \overline{C}_{\alpha f} &= \mathbf{\Gamma_D}\left(C_{\alpha f}, e, w\right) = C_{\alpha f} \cdot \prod_{i=1}^{3} w_i^{r_{C_{\alpha f}}(i)} \\ &= C_{\alpha f} \cdot \left(m^{r_{C_{\alpha f}}(1)} \cdot L^{r_{C_{\alpha f}}(2)} \cdot U^{r_{C_{\alpha f}}(3)}\right) \\ &= C_{\alpha f} \cdot \left(m^{-1} \cdot L^{1} \cdot U^{-2}\right) = C_{\alpha f} \cdot \frac{L}{mU^2}\end{aligned}$$

Example 3: The mass moment of inertia, $I_z$:

$$r_{I_z} = -d_{I_z,e}^T \cdot A_D^{-T} = -\begin{bmatrix} 2 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & -1 \end{bmatrix} = \begin{bmatrix} -1 & -2 & 0 \end{bmatrix}$$

hence,

$$\begin{aligned}\pi_5 = \overline{I}_z &= \mathbf{\Gamma_D}\left(I_z, e, w\right) = I_z \cdot \prod_{i=1}^{3} w_i^{r_{C_{\alpha f}}(i)} \\ &= I_z \cdot \left(m^{r_{I_z}(1)} \cdot L^{r_{I_z}(2)} \cdot U^{r_{I_z}(3)}\right) \\ &= I_z \cdot \left(m^{-1} \cdot L^{-2} \cdot U^{0}\right) = I_z \cdot \frac{1}{mL^2}\end{aligned}$$

The expressions of all the transformed variables of the bicycle model are summarized in Table **6-2**.

Table **6-2**: Summary of the dimensionless parameters and signals of the bicycle model

<table>
<tr><th>Parameters</th><th>Signals</th></tr>
<tr><td rowspan="2">$\pi_1 = \frac{1}{L} \cdot a$<br>$\pi_2 = \frac{1}{L} \cdot b$<br>$\pi_3 = \frac{L}{m \cdot U^2} \cdot C_{\alpha f}$<br>$\pi_4 = \frac{L}{m \cdot U^2} \cdot C_{\alpha r}$<br>$\pi_5 = \frac{1}{m \cdot L^2} \cdot I_z$</td><td>$\pi_6 = \delta_f = \overline{\delta}_f$<br>$\pi_7 = \psi = \overline{\psi}$<br>$\pi_8 = \frac{1}{L} \cdot y = \overline{y}$</td></tr>
<tr><td>Time<br>$\pi_9 = \frac{U}{L} \cdot t = \tau$</td></tr>
</table>

The first five ($\pi_1 \ldots \pi_5$) are the systems parameters expressed in dimensionless form while the remaining dimensionless variables ($\pi_6 \ldots \pi_9$) are the signals which are used in describing the dynamics of the system. These signals include states, inputs and the independent variable time.

#### 6.3.2.3 Dimensional Transformation of a Variable with Time Rate, $\mathbf{\Gamma_D}$

The dimensional transformation of a variable with a time derivative is slightly different from the normal variable because the derivative operator is time dependent. The definition of a parameter transformation, therefore, needs to be extended further to generalize to differentiation (or integration) operators as well as parameters to include inherent time dependence. If the system description has a unity time scaling, there is no need to modify the above definition.

*Definition 6.2: The dimensional transformation of variable with a time rate,* $\Gamma_D$*:* Given a variable $v$, a unit systems $e$ and the scaling variables $\begin{bmatrix} w_1 & w_2 & \cdots & w_n \end{bmatrix}$, the dimensional transformation of $q$ repeated derivatives of $v$ with respect to time, i.e. $\frac{d^q}{dt^q}(v)$, to its corresponding dimensionless quantity $\frac{d^q}{d\tau^q}(\bar{v})$ with new time scaling ($\tau = \beta \cdot t$), is defined by Eq. **6.10**.

$$\frac{d^q}{d\tau^q}(\bar{v}) = \mathbf{\Gamma_D}\left(\frac{d^q}{dt^q}(v), e, w\right) = \beta^{-q} \cdot \frac{d^q}{dt^q}(v) \cdot \prod_{i=1}^{n} w_i^{r_v(i)} = \beta^{-q} \cdot \frac{d^q}{dt^q}\left(\mathbf{\Gamma_D}(v, e, w)\right) \qquad \mathbf{6.10}$$

$$\text{with, } r_v = -d_{v,e}^T \cdot A_D^{-T}$$

If $q = 0$ or $\beta = 1$, then Eq. **6.10** reduces to Eq. **6.8** as one would expect. As an example, consider the second state of the bicycle model $\dot{y}$ in which case $q = 1$ (refer to Eq. **6.2**). From Table **6-2**,

$$\tau = \frac{U}{L} \cdot t \quad \Rightarrow \quad \beta = \frac{U}{L}$$

also,

$$\bar{y} = \mathbf{\Gamma_D}(y, e, w) = \frac{1}{L} \cdot y$$

hence,

$$\begin{aligned} \frac{d}{d\tau}(\bar{y}) = \bar{y}' = \mathbf{\Gamma_D}\left(\frac{d}{dt}(y), e, w\right) &= \beta^{-1} \cdot \frac{d}{dt}(\mathbf{\Gamma_D}(y, e, w)) \\ &= \frac{L}{U} \cdot \frac{d}{dt}\left(\frac{1}{L} \cdot y\right) = \frac{1}{U} \cdot \dot{y} \end{aligned}$$

The above transformations are all invertible, a very important property that is required for system transformation. With the transformations of derivatives and parameters defined thus far, one can now transform a system dynamic model from a dimensional representation to dimensionless and vice versa.

#### 6.3.2.4 The Dimensional Transformation of Systems, $\Im_\mathbf{D}$

The dimensional transformation of systems from/to dimensional to/from dimensionless is a two step process. First the state vector, input and output vectors are transformed to determine the system transformation matrices, i.e., state, input and output vectors transformation. Then the state transformation matrices are used to transform the system from dimensionless to dimensional and vice versa. The state vector transformation can be determined from variable transformation defined in Eq. **6.10**. Consider, for example, the five state vector:

$$\mathbf{x} = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 & x_5 \end{bmatrix}^T = \begin{bmatrix} x & \dot{x} & \ddot{x} & y & \dot{y} \end{bmatrix}^T \qquad \textbf{6.11}$$

The transformation of the vector to dimensionless can be determined as,

$$\bar{\mathbf{x}} = \mathbf{\Gamma_D}(\mathbf{x}, e, w) = \begin{bmatrix} \mathbf{\Gamma_D}(x_1, e, w) \\ \mathbf{\Gamma_D}(x_2, e, w) \\ \mathbf{\Gamma_D}(x_3, e, w) \\ \mathbf{\Gamma_D}(x_4, e, w) \\ \mathbf{\Gamma_D}(x_5, e, w) \end{bmatrix} = \begin{bmatrix} \mathbf{\Gamma_D}(x, e, w) \\ \mathbf{\Gamma_D}\left(\frac{d}{dt} x, e, w\right) \\ \mathbf{\Gamma_D}\left(\frac{d^2}{dt^2} x, e, w\right) \\ \mathbf{\Gamma_D}(y, e, w) \\ \mathbf{\Gamma_D}\left(\frac{d}{dt} y, e, w\right) \end{bmatrix} = \begin{bmatrix} \mathbf{\Gamma_D}(x, e, w) \\ \beta^{-1} \cdot \frac{d}{dt}\left(\mathbf{\Gamma_D}(x, e, w)\right) \\ \beta^{-2} \cdot \frac{d^2}{dt^2}\left(\mathbf{\Gamma_D}(x, e, w)\right) \\ \mathbf{\Gamma_D}(y, e, w) \\ \beta^{-1} \cdot \frac{d}{dt}\left(\mathbf{\Gamma_D}(y, e, w)\right) \end{bmatrix}$$

$$\Rightarrow \bar{\mathbf{x}} = \begin{bmatrix} \prod_{i=1}^{n} w_i^{r_x(i)} & 0 & 0 & 0 & 0 \\ 0 & \beta^{-1} \cdot \prod_{i=1}^{n} w_i^{r_x(i)} & 0 & 0 & 0 \\ 0 & 0 & \beta^{-2} \cdot \prod_{i=1}^{n} w_i^{r_x(i)} & 0 & 0 \\ 0 & 0 & 0 & \prod_{i=1}^{n} w_i^{r_y(i)} & 0 \\ 0 & 0 & 0 & 0 & \beta^{-1} \cdot \prod_{i=1}^{n} w_i^{r_y(i)} \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \\ y \\ \dot{y} \end{bmatrix}$$

therefore, the state transformation, Eq. **6.12**, is a linear transformation accounting for parametric scaling and time scaling. It is more conveniently and compactly defined as:

$$\bar{\mathbf{x}} = \mathbf{M_x x} \qquad \textbf{6.12}$$

where, for the above state vector $\mathbf{x}=\begin{bmatrix}x & \dot{x} & \ddot{x} & y & \dot{y}\end{bmatrix}^T$ for example, $\mathbf{M_x}$ is given by Eq. **6.13** and for any other state vectors it can be determined in the same way.

$$\mathbf{M_x}=\begin{bmatrix}\prod_{i=1}^{n} w_i^{r_x(i)} & 0 & 0 & 0 & 0 \\ 0 & \beta^{-1}\cdot\prod_{i=1}^{n} w_i^{r_x(i)} & 0 & 0 & 0 \\ 0 & 0 & \beta^{-2}\cdot\prod_{i=1}^{n} w_i^{r_x(i)} & 0 & 0 \\ 0 & 0 & 0 & \prod_{i=1}^{n} w_i^{r_y(i)} & 0 \\ 0 & 0 & 0 & 0 & \beta^{-1}\cdot\prod_{i=1}^{n} w_i^{r_y(i)}\end{bmatrix} \qquad \mathbf{6.13}$$

The transformation for output vectors is similar to that of the state vectors. For example, consider the output vector $\mathbf{y}=\begin{bmatrix}y_1 & y_2 & y_3\end{bmatrix}^T=\begin{bmatrix}x & \dot{x} & y\end{bmatrix}^T$. The transformation of the vector to dimensionless can be determined as,

$$\bar{\mathbf{y}} = \mathbf{\Gamma_D}(\mathbf{y},e,w)=\begin{bmatrix}\mathbf{\Gamma_D}(y_1,e,w) \\ \mathbf{\Gamma_D}(y_2,e,w) \\ \mathbf{\Gamma_D}(y_3,e,w)\end{bmatrix}=\begin{bmatrix}\mathbf{\Gamma_D}(x,e,w) \\ \mathbf{\Gamma_D}\left(\frac{d}{dt}x,e,w\right) \\ \mathbf{\Gamma_D}(y,e,w)\end{bmatrix}=\begin{bmatrix}\mathbf{\Gamma_D}(x,e,w) \\ \beta^{-1}\cdot\frac{d}{dt}\left(\mathbf{\Gamma_D}(x,e,w)\right) \\ \mathbf{\Gamma_D}(y,e,w)\end{bmatrix}$$

$$\Rightarrow \quad \bar{\mathbf{y}} = \begin{bmatrix}\prod_{i=1}^{n} w_i^{r_x(i)} & 0 & 0 \\ 0 & \beta^{-1}\cdot\prod_{i=1}^{n} w_i^{r_x(i)} & 0 \\ 0 & 0 & \prod_{i=1}^{n} w_i^{r_y(i)}\end{bmatrix}\begin{bmatrix}x \\ \dot{x} \\ y\end{bmatrix}$$

In the same fashion, the transformation of the output vector, Eq. **6.14**, is a linear transformation accounting for parametric scaling and time scaling of the output signals.

$$\bar{\mathbf{y}} = \mathbf{M_y}\mathbf{y} \qquad \mathbf{6.14}$$

where, for the above output vector $\mathbf{y} = \begin{bmatrix} x & \dot{x} & y \end{bmatrix}^T$, $\mathbf{M}_\mathbf{y}$ is given by Eq. **6.15** and for any other output vectors it can be determined in a similar fashion.

$$\mathbf{M}_\mathbf{y} = \begin{bmatrix} \prod_{i=1}^{n} w_i^{r_x(i)} & 0 & 0 \\ 0 & \beta^{-1} \cdot \prod_{i=1}^{n} w_i^{r_x(i)} & 0 \\ 0 & 0 & \prod_{i=1}^{n} w_i^{r_y(i)} \end{bmatrix} \qquad \mathbf{6.15}$$

Finally, the transformation of the control input vector is performed in the same way. As an example, consider the input vector $\mathbf{u} = \begin{bmatrix} u_1 & u_2 & u_3 \end{bmatrix}^T$, the transformation of the vector to dimensionless can be determined as:

$$\overline{\mathbf{u}} = \mathbf{\Gamma}_\mathbf{D}(\mathbf{u}, e, w) = \begin{bmatrix} \mathbf{\Gamma}_\mathbf{D}(u_1, e, w) \\ \mathbf{\Gamma}_\mathbf{D}(u_2, e, w) \\ \mathbf{\Gamma}_\mathbf{D}(u_3, e, w) \end{bmatrix}$$

$$\Rightarrow \quad \overline{\mathbf{u}} = \begin{bmatrix} \prod_{i=1}^{n} w_i^{r_{u1}(i)} & 0 & 0 \\ 0 & \prod_{i=1}^{n} w_i^{r_{u2}(i)} & 0 \\ 0 & 0 & \prod_{i=1}^{n} w_i^{r_{u3}(i)} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$$

The input transformation, defined in Eq. **6.16** , is also a linear transformation.

$$\overline{\mathbf{u}} = \mathbf{M}_\mathbf{u} \mathbf{u} \qquad \mathbf{6.16}$$

where, for the above input vector $\mathbf{u} = \begin{bmatrix} u_1 & u_2 & u_3 \end{bmatrix}^T$, $\mathbf{M}_\mathbf{u}$ is given by Eq. **6.17** and for any other output vectors it can be determined in a similarly.

$$\mathbf{M_u} = \begin{bmatrix} \prod_{i=1}^{n} w_i^{r_{u1}(i)} & 0 & 0 \\ 0 & \prod_{i=1}^{n} w_i^{r_{u2}(i)} & 0 \\ 0 & 0 & \prod_{i=1}^{n} w_i^{r_{u3}(i)} \end{bmatrix} \qquad \mathbf{6.17}$$

In the above example time derivatives of the control input are considered because in practical problems it is not common to have a time derivative of input signals. However, Eq. **6.16** is general and when there are applications that require the time derivative of a control input as part of the input vector, then the input transformation matrix, $\mathbf{M_u}$, is determined similar to that of the state and output vectors.

The transformation of systems (dynamic models), $\Im_{\mathbf{D}}$, that are represented in state-space form are discussed next. Consider a general plant model $G \equiv \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]$ expressed in the dimensional domain with $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{u}$ as the state vector, output vector and input vector, respectively. The systems can be transformed to an equivalent representation, $\overline{G} \equiv \left[\begin{array}{c|c} \overline{A} & \overline{B} \\ \hline \overline{C} & \overline{D} \end{array}\right]$ expressed in the dimensionless domain with $\overline{\mathbf{x}}$, $\overline{\mathbf{y}}$ and $\overline{\mathbf{u}}$ as the new state, output and input vectors, respectively. Expressing, $G$ in the regular state-space form, Eq. **6.18**, the state, output and input vectors can be transformed to the dimensionless domain using the dimensional transformation operator, $\overline{\mathbf{v}} = \mathbf{\Gamma_D}(\mathbf{v}, e, w)$.

$$\begin{aligned} \dot{\mathbf{x}} &= A\mathbf{x} + B\mathbf{u} \\ \mathbf{y} &= C\mathbf{x} + D\mathbf{u} \end{aligned} \qquad \mathbf{6.18}$$

Applying the operator to the state, output and input vectors, Eq. **6.12**, Eq. **6.14** and, Eq. **6.16**, and solving for $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{u}$ yields the inverse transformation relation of the vectors given in Eq. **6.19**.

$$\begin{aligned} \bar{\mathbf{x}} &= \mathbf{M_x x} \quad \Rightarrow \quad \mathbf{x} = \mathbf{M_x^{-1}}\bar{\mathbf{x}} \\ \bar{\mathbf{y}} &= \mathbf{M_y y} \quad \Rightarrow \quad \mathbf{y} = \mathbf{M_y^{-1}}\bar{\mathbf{y}} \\ \bar{\mathbf{u}} &= \mathbf{M_u u} \quad \Rightarrow \quad \mathbf{u} = \mathbf{M_u^{-1}}\bar{\mathbf{u}} \end{aligned} \qquad \textbf{6.19}$$

The transformation of the time rate of the state vector, Eq. **6.20**, is determined using Eq. **6.10**, applied to a vector, i.e.,

$$\begin{aligned} \frac{d}{d\tau}(\bar{\mathbf{x}}) &= \bar{\mathbf{x}}' = \beta^{-1}\cdot\frac{d}{dt}\left(\mathbf{\Gamma_D}(\mathbf{x},u,w)\right) \\ &= \beta^{-1}\cdot\frac{d}{dt}(\mathbf{M_x x}) \\ &\Rightarrow \quad \bar{\mathbf{x}}' = \beta^{-1}\cdot\mathbf{M_x}\dot{\mathbf{x}} \end{aligned} \qquad \textbf{6.20}$$

solving for $\dot{\mathbf{x}}$ yields Eq. **6.21**

$$\bar{\mathbf{x}}' = \beta^{-1}\cdot\mathbf{M_x}\dot{\mathbf{x}} \quad \Rightarrow \quad \dot{\mathbf{x}} = \beta\cdot\mathbf{M_x^{-1}}\bar{\mathbf{x}}' \qquad \textbf{6.21}$$

substituting Eq. **6.19** and Eq. **6.21** into Eq. **6.18**, yields Eq. **6.22**

$$\begin{aligned} \beta\cdot\mathbf{M_x^{-1}}\bar{\mathbf{x}}' &= A\mathbf{M_x^{-1}}\bar{\mathbf{x}} + B\mathbf{M_u^{-1}}\bar{\mathbf{u}} \\ \mathbf{M_y^{-1}}\bar{\mathbf{y}} &= C\mathbf{M_x^{-1}}\bar{\mathbf{x}} + D\mathbf{M_u^{-1}}\bar{\mathbf{u}} \\ \Rightarrow \quad \bar{\mathbf{x}}' &= \beta^{-1}\cdot\mathbf{M_x}A\mathbf{M_x^{-1}}\bar{\mathbf{x}} + \beta^{-1}\cdot\mathbf{M_x}B\mathbf{M_u^{-1}}\bar{\mathbf{u}} \\ \bar{\mathbf{y}} &= \mathbf{M_y}C\mathbf{M_x^{-1}}\bar{\mathbf{x}} + \mathbf{M_y}D\mathbf{M_u^{-1}}\bar{\mathbf{u}} \end{aligned} \qquad \textbf{6.22}$$

Therefore, a dimensionless plant can be expressed using the state-space form of Eq. **6.23**.

$$\begin{aligned} \bar{\mathbf{x}}' &= \bar{A}\bar{\mathbf{x}} + \bar{B}\bar{\mathbf{u}} \\ \bar{\mathbf{y}} &= \bar{C}\bar{\mathbf{x}} + \bar{D}\bar{\mathbf{u}} \end{aligned} \qquad \textbf{6.23}$$

with the dimensionless system matrices expressed in Eq. **6.24**

$$\begin{aligned}\bar{A} &= \beta^{-1} \cdot \mathbf{M_x} A \mathbf{M_x}^{-1} \\ \bar{B} &= \beta^{-1} \cdot \mathbf{M_x} B \mathbf{M_u}^{-1} \\ \bar{C} &= \mathbf{M_y} C \mathbf{M_x}^{-1} \\ \bar{D} &= \mathbf{M_y} D \mathbf{M_u}^{-1}\end{aligned} \qquad \textbf{6.24}$$

The formal definition of the dimensional transformation of a system, expressed in state-space form is given in Eq. **6.25**, in terms of the transformation of the systems matrices. Similar transformation operator can be developed for systems expressed as transfer functions.

*Definition 6.3: The dimensional transformation of a system,* $\Im_D$ *:* Given a general plant model $G \equiv \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]$ expressed in the dimensional domain with $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{u}$ as the state vector, output vector and input vector, respectively. The dimensional transformation to an equivalent representation, $\bar{G} \equiv \left[\begin{array}{c|c} \bar{A} & \bar{B} \\ \hline \bar{C} & \bar{D} \end{array}\right]$ expressed in the dimensionless domain with $\bar{\mathbf{x}}$, $\bar{\mathbf{y}}$, $\bar{\mathbf{u}}$ and $\beta$ as the new state vector, output vector, input vector and a new time scaling (i.e., $\tau = \beta \cdot t$), respectively, is as defined as:

$$\begin{aligned}\bar{G} &= \left[\begin{array}{c|c} \bar{A} & \bar{B} \\ \hline \bar{C} & \bar{D} \end{array}\right] = \Im_{\mathbf{D}}\left(G, \mathbf{M_x}, \mathbf{M_y}, \mathbf{M_u}, \beta\right) \\ &= \left[\begin{array}{c|c} \beta^{-1} \cdot \mathbf{M_x} A \mathbf{M_x}^{-1} & \beta^{-1} \cdot \mathbf{M_x} B \mathbf{M_u}^{-1} \\ \hline \mathbf{M_y} C \mathbf{M_x}^{-1} & \mathbf{M_y} D \mathbf{M_u}^{-1} \end{array}\right]\end{aligned} \qquad \textbf{6.25}$$

such that,

$$\begin{aligned}\bar{\mathbf{x}}' &= \bar{A}\bar{\mathbf{x}} + \bar{B}\bar{\mathbf{u}} \\ \bar{\mathbf{y}} &= \bar{C}\bar{\mathbf{x}} + \bar{D}\bar{\mathbf{u}}\end{aligned}$$

where, $\mathbf{M_x}$, $\mathbf{M_y}$ and $\mathbf{M_u}$ are the state, output and input transformation matrices.

The system transformation for a control system used with a plant is derived from the transformation of the plant used in designing the controller. The robust controller synthesis in general requires adding weights on selected signals to meet the stability and/or performance requirements. Since the order of the robust controller is generally equal to the plant order plus the order of all performance weights used, addition of signal weight increases the order of the controller. Therefore, the transformation is separately studied and finally generalized to a control system with or without performance and/or stability weights. Consider a robust control system that has the same order as the plant. The plant-controller interconnection of Figure **6-3** , shows the output of the plant is an input to the controller and vice versa. The state-space description of the robust controller $K$ of Eq. **6.26** , is assumed to have the same order as the plant $G$ .

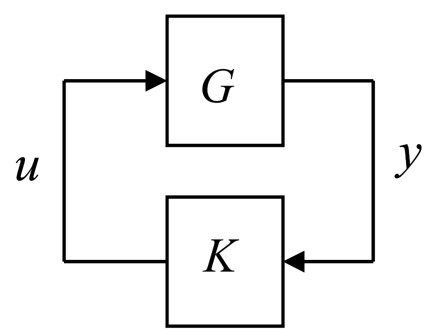



Figure **6-3**: Plant-controller feedback interconnection

$$\begin{aligned}\dot{\mathbf{x}}_k &= A_k\mathbf{x}_k + B_k\mathbf{y}\\ \mathbf{u} &= C_k\mathbf{x}_k + D_k\mathbf{y}\end{aligned} \qquad \mathbf{6.26}$$

The controller state vector $\mathbf{x}_k$ for a robust controller generally represents the same physical quantity as the plant state vector $\mathbf{x}$, i.e., if the first element of $\mathbf{x}$ represents displacement, then the first element of the controller state vector $\mathbf{x}_k$ also represent displacement. If the second element of the state vector $\mathbf{x}$ refers to speed, so does the second element of the controller state vector $\mathbf{x}_k$. This is always true regardless of the reference of measurement. For example, in an estimator type of dynamic controller, the controller state vector is usually an error vector defined as the difference between the actual and estimated states of the plant. Robust controllers also have a similar form. Therefore, the dimensional transformation of the controller state vector, Eq. **6.27** , is similar to the plant state vector when there are no performance weights involved.

$$\bar{\mathbf{x}}_k = \mathbf{M}_{\mathbf{x}}\mathbf{x}_k \qquad \textbf{6.27}$$

solving for $\mathbf{x}_k$, yields

$$\bar{\mathbf{x}}_k = \mathbf{M}_{\mathbf{x}}\mathbf{x}_k \quad \Rightarrow \quad \mathbf{x}_k = \mathbf{M}_{\mathbf{x}}^{-1}\bar{\mathbf{x}}_k \qquad \textbf{6.28}$$

The dimensional transformation of the time rate of the controller state vector is also determined using Eq. **6.10** applied to a vector as

$$\begin{aligned} \frac{d}{d\tau}\left(\bar{\mathbf{x}}_k\right) &= \bar{\mathbf{x}}_k' = \beta^{-1}\cdot\frac{d}{dt}\left(\mathbf{\Gamma}_{\mathbf{D}}\left(\mathbf{x}_k, u, w\right)\right) \\ &= \beta^{-1}\cdot\frac{d}{dt}\left(\mathbf{M}_{\mathbf{x}}\mathbf{x}_k\right) \\ \Rightarrow \quad \bar{\mathbf{x}}_k' &= \beta^{-1}\cdot\mathbf{M}_{\mathbf{x}}\dot{\mathbf{x}}_k \end{aligned} \qquad \textbf{6.29}$$

Solving for $\dot{\mathbf{x}}_k$

$$\bar{\mathbf{x}}_k' = \beta^{-1}\cdot\mathbf{M}_{\mathbf{x}}\dot{\mathbf{x}}_k \quad \Rightarrow \quad \dot{\mathbf{x}}_k = \beta\cdot\mathbf{M}_{\mathbf{x}}^{-1}\bar{\mathbf{x}}_k' \qquad \textbf{6.30}$$

Substituting the solutions of $\mathbf{y}$ and $\mathbf{u}$ from Eq. **6.19**, solution of $\mathbf{x}_k$ and $\dot{\mathbf{x}}_k$ from Eq. **6.28** and Eq. **6.30**, respectively into Eq. **6.26**,

$$\begin{aligned} \beta \cdot \mathbf{M}_{\mathbf{x}}^{-1}\overline{\mathbf{x}}_k' &= A_k \mathbf{M}_{\mathbf{x}}^{-1}\overline{\mathbf{x}}_k + B_k \mathbf{M}_{\mathbf{y}}^{-1}\overline{\mathbf{y}} \\ \mathbf{M}_{\mathbf{u}}^{-1}\overline{\mathbf{u}} &= C_k \mathbf{M}_{\mathbf{x}}^{-1}\overline{\mathbf{x}}_k + D_k \mathbf{M}_{\mathbf{y}}^{-1}\overline{\mathbf{y}} \end{aligned}$$

$$\Rightarrow \begin{aligned} \overline{\mathbf{x}}_k' &= \beta^{-1} \cdot \mathbf{M}_{\mathbf{x}} A_k \mathbf{M}_{\mathbf{x}}^{-1}\overline{\mathbf{x}}_k + \beta^{-1} \cdot \mathbf{M}_{\mathbf{x}} B_k \mathbf{M}_{\mathbf{y}}^{-1}\overline{\mathbf{y}} \\ \overline{\mathbf{u}} &= \mathbf{M}_{\mathbf{u}} C_k \mathbf{M}_{\mathbf{x}}^{-1}\overline{\mathbf{x}}_k + \mathbf{M}_{\mathbf{u}} D_k \mathbf{M}_{\mathbf{y}}^{-1}\overline{\mathbf{y}} \end{aligned}$$

Therefore, dimensionless controller is expressed in state-space as

$$\begin{aligned} \overline{\mathbf{x}}_k' &= \overline{A}_k \overline{\mathbf{x}}_k + \overline{B}\overline{\mathbf{y}} \\ \overline{\mathbf{u}} &= \overline{C}_k \overline{\mathbf{x}} + \overline{D}_k \overline{\mathbf{y}} \end{aligned} \qquad \mathbf{6.31}$$

with the dimensionless controller system matrices expressed in Eq. **6.32**

$$\begin{aligned} \overline{A}_k &= \beta^{-1} \cdot \mathbf{M}_{\mathbf{x}} A_k \mathbf{M}_{\mathbf{x}}^{-1} \\ \overline{B}_k &= \beta^{-1} \cdot \mathbf{M}_{\mathbf{x}} B_k \mathbf{M}_{\mathbf{y}}^{-1} \\ \overline{C}_k &= \mathbf{M}_{\mathbf{u}} C_k \mathbf{M}_{\mathbf{x}}^{-1} \\ \overline{D}_k &= \mathbf{M}_{\mathbf{u}} D_k \mathbf{M}_{\mathbf{y}}^{-1} \end{aligned} \qquad \mathbf{6.32}$$

With the above results, the dimensional transformation of a control system is formally defined in terms of the dimensional transformation matrices of the plant system transformation matrices.

*Definition 6.4: The dimensional transformation of a control system,* $\Im_D$ *:* Given a general plant model $G \equiv \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]$ described in the dimensional domain with $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{u}$ as the state vector, output vector and input vector, respectively. Also, given a feedback control system $K \equiv \left[\begin{array}{c|c} A_k & B_k \\ \hline C_k & D_k \end{array}\right]$ described in the dimensional domain with $\mathbf{x}_k$, $\mathbf{u}$ and $\mathbf{y}$ as the controller state, controller output and controller input vectors, respectively. If $G$ is dimensionally transformation to an equivalent representation, $\bar{G} \equiv \left[\begin{array}{c|c} \bar{A} & \bar{B} \\ \hline \bar{C} & \bar{D} \end{array}\right]$ expressed in the dimensionless domain with $\bar{\mathbf{x}}$, $\bar{\mathbf{y}}$, $\bar{\mathbf{u}}$ and $\beta$ as the new state vector, output vector, input vector and a new time scaling (i.e., $\tau = \beta \cdot t$), respectively, using the dimensional transformation matrices $\mathbf{M_x}$, $\mathbf{M_y}$ and $\mathbf{M_u}$, then $K$ can be transformed to the equivalent form $\bar{K} \equiv \left[\begin{array}{c|c} \bar{A}_k & \bar{B}_k \\ \hline \bar{C}_k & \bar{D}_k \end{array}\right]$ expressed in the dimensionless domain with $\bar{\mathbf{x}}_k$, $\bar{\mathbf{u}}$ and $\bar{\mathbf{y}}$ as the new controller state vector, controller output vector and controller input vector, respectively. This transformation is mathematically stated in Eq. **6.33** .

$$\begin{aligned} \bar{K} &= \left[\begin{array}{c|c} \bar{A}_k & \bar{B}_k \\ \hline \bar{C}_k & \bar{D}_k \end{array}\right] = \Im_{\mathbf{D}}\left(K, \mathbf{M_x}, \mathbf{M_y}, \mathbf{M_u}, \beta\right) \\ &= \left[\begin{array}{c|c} \beta^{-1} \cdot \mathbf{M_x} A_k \mathbf{M_x}^{-1} & \beta^{-1} \cdot \mathbf{M_x} B_k \mathbf{M_y}^{-1} \\ \hline \mathbf{M_u} C_k \mathbf{M_x}^{-1} & \mathbf{M_u} D_k \mathbf{M_y}^{-1} \end{array}\right] \end{aligned} \qquad \mathbf{6.33}$$

such that,

$$\begin{aligned} \bar{\mathbf{x}}_k' &= \bar{A}_k \bar{\mathbf{x}}_k + \bar{B}\bar{\mathbf{y}} \\ \bar{\mathbf{u}} &= \bar{C}_k \bar{\mathbf{x}} + \bar{D}_k \bar{\mathbf{y}} \end{aligned}$$

The control system transformation of Eq. **6.33**, assumes that the controller has the same order as the plant model and the states of the controller belong to the same physical quantity as the states of the plant. The case when a control system has higher order than the plant model due to the signal weights added during the robust control synthesis and discussed next.

Performance weights are generally frequency-dependent penalty functions imposed on signals to force the signals to be have expected characteristics. For example higher control actions can be penalized by putting weights that have lower magnitude at low frequencies and larger magnitude at higher frequencies on the control signal. One of the challenges, at least in this thesis, in generalizing the dimensional transformation to higher order performance weights is due to the practice that performance weights are described by their transfer functions and the corresponding state space representation is non-unique. However, lower order examples are used to explain how such weights can be incorporated into the transformation.

***Example 1***: The simplest signal weight is a constant. This is a zero order weight and therefore, does not affect the order of the plant.

***Example 2***: First order weight adds one order to the robust controller. A first order system has a single state. Therefore, its scaling is unity and the state transformation matrix in this case is given by

$$\mathbf{M}_{\mathbf{xkw}} = \begin{bmatrix} 1 \end{bmatrix} \qquad \textbf{6.34}$$

***Example 3***: Second order weight adds two more orders to the robust controller. In this case there are two options:

1) If the system has single degree of freedom then the second state of this system is a time derivative of the first state and the state transformation matrix is given by Eq. **6.35**.

$$\mathbf{M}_{\mathbf{xkw}} = \begin{bmatrix} 1 & 0 \\ 0 & \beta^{-1} \end{bmatrix} \qquad \textbf{6.35}$$

2) If the system has two degrees of freedom, then both states are coupled first order system of equations and the state transformation matrix becomes an identity as

$$\mathbf{M}_{\mathbf{xkw}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad \mathbf{6.36}$$

For higher-order performance weights it is not straight forward to determine which realization is used to convert the transfer function to state-space representation of the system.

Once the state transformation of the performance weights is determined, the controller state transformation is determined from the plant state transformation matrix and the state transformation matrix of the performance weights using Eq. **6.37**

$$\mathbf{M}_{\mathbf{xk}} = \begin{bmatrix} \mathbf{M}_{\mathbf{x}} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_{\mathbf{xkw}} \end{bmatrix} \qquad \mathbf{6.37}$$

where $\mathbf{M}_{\mathbf{xk}}$, $\mathbf{M}_{\mathbf{x}}$ and $\mathbf{M}_{\mathbf{xkw}}$ are the state transformation matrices of the control state vector, plant state vector and state vector of the additional weights, respectively.

### 6.3.2.5 Controllability and Observability of the Dimensionless Representation

The concepts of controllability and observability of a system are important conditions to solve feedback control and observer problems. This discusses the invariance of the controllability and observability conditions under the dimensional transformation. The controllability and observability conditions of state-space systems are given below.

***Controllability***: in the state-space system, $G \equiv \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]$, the pair $(A, B)$ is completely controllable if and only if the controllability matrix $Q_C$ given by Eq. **6.38** has rank $n$, or equivalently $Q_C \cdot Q_C^T$ is invertible.

$$Q_C = \begin{bmatrix} B & AB & A^2B & \cdots & A^{n-1}B \end{bmatrix} \qquad \textbf{6.38}$$

***Observability***: in the state-space system, $G \equiv \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]$, the pair $(A, C)$ is completely observable if and only if the observability matrix $Q_O$ given by Eq. **6.39** has rank $n$, or equivalently $Q_O^T \cdot Q_O$ is invertible.

$$Q_O = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix} \qquad \textbf{6.39}$$

***Theorem 6.1***: the controllability and observability conditions of a system are maintained under dimensional transformation, i.e., if a systems is controllable and/or observable in the dimensional domain then its dimensionless representation is also controllable and/or observable and vice-versa.

***Proof***: recall that the system matrices of the dimensionless system are expressed in terms of the system matrices of the dimensional system, the transformation matrices and the time scaling as:

$$\begin{aligned}
\bar{A} &= \beta^{-1} \cdot \mathbf{M_x} A \mathbf{M_x}^{-1} \\
\bar{B} &= \beta^{-1} \cdot \mathbf{M_x} B \mathbf{M_u}^{-1} \\
\bar{C} &= \mathbf{M_y} C \mathbf{M_x}^{-1} \\
\bar{D} &= \mathbf{M_y} D \mathbf{M_u}^{-1}
\end{aligned}$$

the matrix $\bar{A}$ raised to the different powers can be expressed as

$$\begin{aligned}
&\bar{A} = \beta^{-1} \cdot \mathbf{M_x} A \mathbf{M_x}^{-1} \\
&\bar{A}^2 = \bar{A} \cdot \bar{A} = \beta^{-1} \cdot \mathbf{M_x} A \mathbf{M_x}^{-1} \cdot \beta^{-1} \cdot \mathbf{M_x} A \mathbf{M_x}^{-1} = \beta^{-2} \cdot \mathbf{M_x} A^2 \mathbf{M_x}^{-1} \\
&\bar{A}^3 = \bar{A}^2 \cdot \bar{A} = \beta^{-2} \cdot \mathbf{M_x} A^2 \mathbf{M_x}^{-1} \cdot \beta^{-1} \cdot \mathbf{M_x} A \mathbf{M_x}^{-1} = \beta^{-3} \cdot \mathbf{M_x} A^3 \mathbf{M_x}^{-1} \\
&\vdots \\
&\bar{A}^k = \beta^{-k} \cdot \mathbf{M_x} A^k \mathbf{M_x}^{-1}
\end{aligned} \qquad \textbf{6.40}$$

Thus, the controllability matrix of the dimensionless system can be determined as Eq. **6.41**.

$$\begin{aligned}
&\bar{Q}_C = \begin{bmatrix} \bar{B} & \bar{A}\bar{B} & \bar{A}^2\bar{B} & \cdots & \bar{A}^{n-1}\bar{B} \end{bmatrix} \\
&\bar{Q}_C = \beta^{-1} \cdot \mathbf{M_x} \begin{bmatrix} B & \beta^{-1}AB & \beta^{-2}A^2B & \cdots & \beta^{-(n-1)}A^{n-1}B \end{bmatrix} \mathbf{M_u}^{-1}
\end{aligned} \qquad \textbf{6.41}$$

if the dimensional system is completely controllable then the controllability matrix

$$Q_C = \begin{bmatrix} B & AB & A^2B & \cdots & A^{n-1}B \end{bmatrix}$$

is full rank. Hence, the matrix given by Eq. **6.42** is also full rank.

$$\begin{bmatrix} B & \beta^{-1}AB & \beta^{-2}A^2B & \cdots & \beta^{-(n-1)}A^{n-1}B \end{bmatrix} \qquad \textbf{6.42}$$

Pre- and post-multiplication of Eq. **6.42** by the full diagonal matrices $\mathbf{M_x}$ and $\mathbf{M_u}^{-1}$, respectively does not change the rank. Therefore, the controllability matrix of the dimensionless system, Eq. **6.41**, is full rank and the dimensionless system is completely controllable.

Similarly, the observability matrix of the dimensionless system is determined as Eq. **6.43**

$$\bar{Q}_O = \begin{bmatrix} \bar{C} \\ \overline{CA} \\ \\ \overline{CA}^{n-1} \end{bmatrix} = \mathbf{M_y} \begin{bmatrix} C \\ \beta^{-1}CA \\ \\ \beta^{-(n-1)}CA^{n-1} \end{bmatrix} \mathbf{M_x^{-1}} \qquad \textbf{6.43}$$

if the dimensional system is completely observable then the observability matrix

$$Q_O = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

is full rank. Thus the matrix given by Eq. **6.44** is also full rank.

$$\begin{bmatrix} C \\ \beta^{-1}CA \\ \\ \beta^{-(n-1)}CA^{n-1} \end{bmatrix} \qquad \textbf{6.44}$$

Pre- and post-multiplication of Eq. **6.44** by the full diagonal matrices $\mathbf{M_y}$ and $\mathbf{M_x^{-1}}$, respectively does not change the rank. Therefore, the observability matrix of the dimensionless system, Eq. **6.43**, is full rank and the dimensionless system is completely observable.

This completes the proof. ■

#### 6.3.2.6 Some General Observation on Vehicle Parameters and Models

The dimensional transformation of the vehicle parameters, Table **6-2**, resulting in the new dimensionless parameters $\pi_1 \ldots \pi_5$ is used to study whether a family of vehicles can be grouped together for a robust control synthesis in the dimensionless domain. The dimensional transformation of vehicle models, pictorially represented in Figure **6-4**, generally result in a lesser model variations, Figure **6-1**, compared to the dimensional vehicle models, because of the coupling nature of some of the vehicle system parameters.

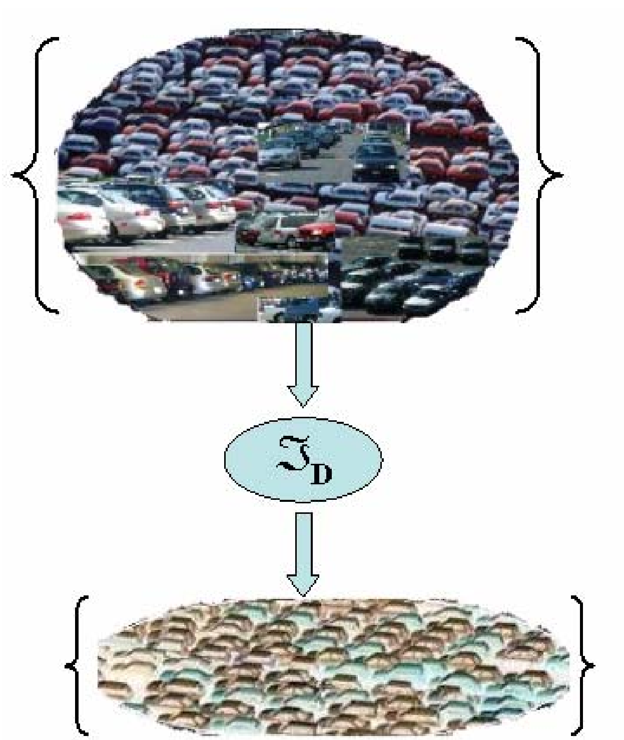
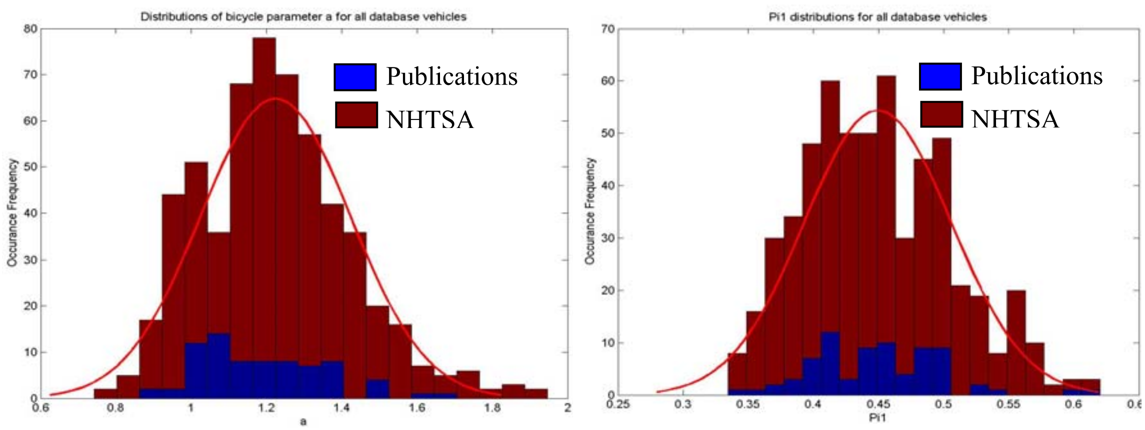



Figure **6-4**: The pictorial dimensional transformation of vehicle systems

A collection of vehicle parameters that are later used to explain the current technique is presented in Figure **6-5** and Figure **6-6**. The vehicle data is collected from difference sources (see [22]), that are broadly categorized into two:

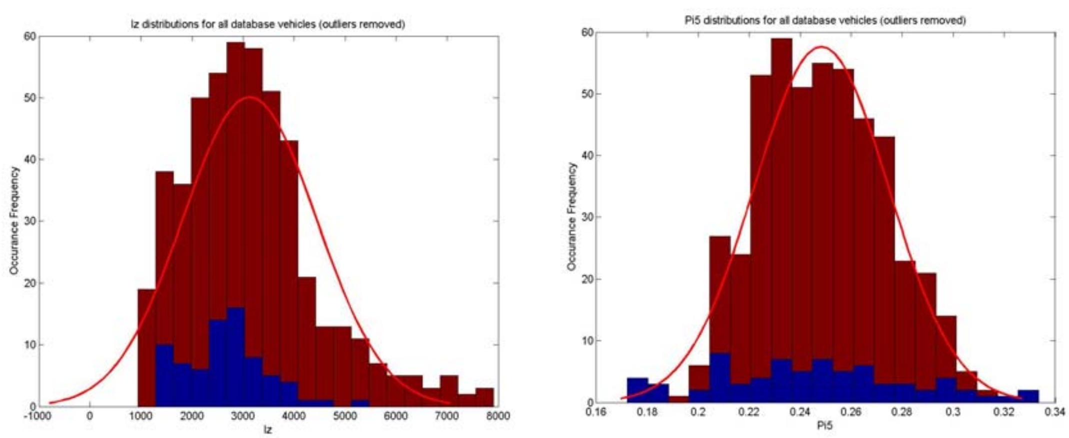
*Publications*: from different publications in vehicle dynamics and control.

*NHTSA*: This is a public database that contains many vehicle parameters.




(a) (b)

Figure **6-5**: Distribution of front axel distance: (a) dimensional, (b) dimensionless




(a) (b)

Figure **6-6**: Distribution of mass moment of inertia, (a) dimensional, (b) dimensionless

### 6.4 The Robust Control Design

The technique developed in this chapter is demonstrated using the vehicle lateral control. This is an automatic steering control called the K-driver and has three main advantages, Figure **6-7**. It is Robust, Adaptive and Modular. The K-driver is robust because the controller synthesis is performed using the robust control synthesis tools to account for model uncertainties. It is adaptive due to the capability to use it in several vehicles through dimensional transformation, an off-line parameter adaptation, with out redesigning the controller. It is also gain-scheduled because some external parameter variations in velocity or cornering stiffness are implicitly cancelled by the use of dimensionless parameters and signal scaling. The controller is modular because its usage is to provide a single module for several vehicles. These vehicles will be denoted with different vehicle identifier (I.D.). The control law is synthesized such that it can automatically identify the parameters from its database corresponding to the specific vehicle and perform the parameter adaptation automatically.

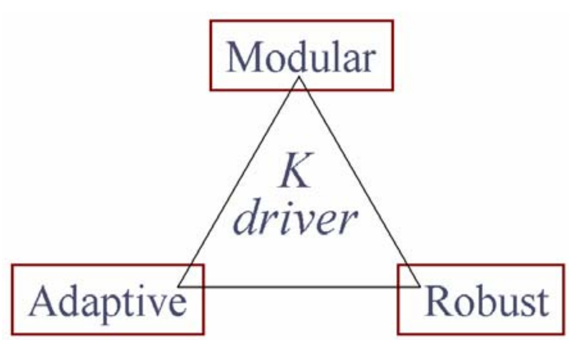



Figure **6-7**: The advantages of the K-driver

The conversion of the models to a more dense representation is performed using dimensional transformation denoted as $\Im_D$. This dimensional transformation $\Im_D$, and its inverse (below) are specific to an individual plant. It is important to note that the individual plant, for example one vehicle, that may be quite dissimilar in size or mass to other plants clustered in the group of systems, all passenger vehicles for example. The conversion process results in a dimensionless plant representation, which when applied to all plants in a group of systems, results in a dimensionless group representation.

A key insight of this scaling approach is that the dimensionless group representation has very little member-to-member variability as compared to the untransformed group. This chapter therefore considers the task of designing a controller for the compact, dimensionless group in aggregate that can be then transformed back to each individual member system as needed.

Once the controller synthesis is performed in dimensionless domain, the controller suitable for the group of systems is transformed to the dimensional domain using the inverse dimensional transformation, $\Im_D^{-1}$ and is ready for implementation on a real system.

In summary the proposed technique involves four steps, schematically shown in Figure **6-8** and are described as follows:

*Step 1 System transformation:* Transform each dimensional model to dimensionless model using the dimensional transformation operator $\Im_D$. At this step it is very important to first identify the system parameters that are strongly coupled, for example how mass and inertia generally change for a given system in a highly constrained manner. It is also equally important to choose the dimensional scaling parameters such that the coupling parameters are grouped together as much as possible.

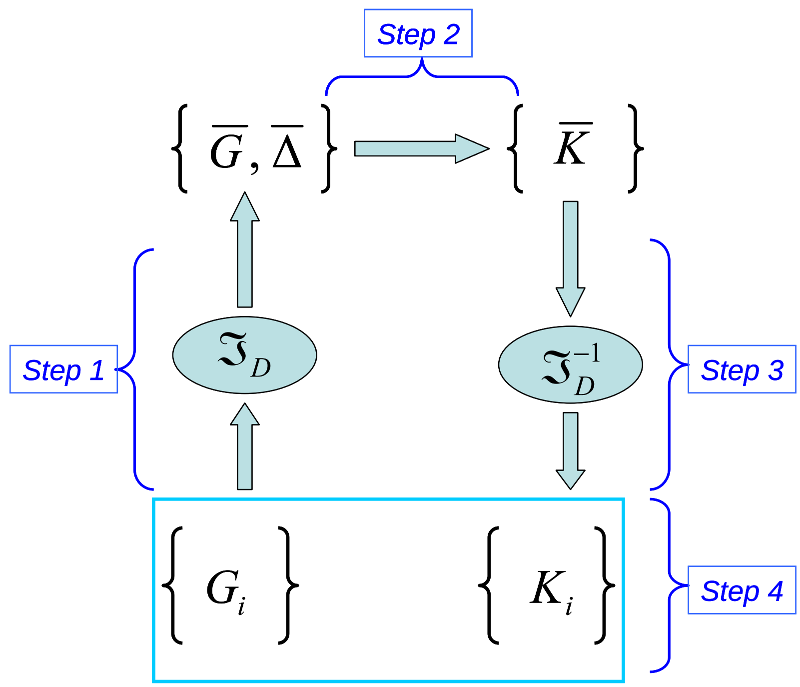



Figure **6-8**: General setup

*Step 2 Perform robust control synthesis:* Define a nominal model and represent the remaining models as model uncertainty. Also determine the stability and performance requirements and perform the robust control synthesis.

*Step 3 Control system transformation:* Transform the dimensionless controller to its corresponding plant using the inverse dimensional transformation operator, $\Im_D^{-1}$.

*Step 4 Verify requirements:* Verify if the controller requirements are all met. If requirement not met go back to *step 2* and repeat the control synthesis with a different signal weights.

The above four steps are explained in a more detail in the following sub-sections using the vehicle lateral control as an example.

### 6.4.1 The LFT representation

The robust control synthesis tools used for this example require the systems to be represented in LFT form. The LFT form, Figure **6-9**, is a representation of the uncertain system with the uncertainty model separated from nominal model and interconnected as a feedback loop.

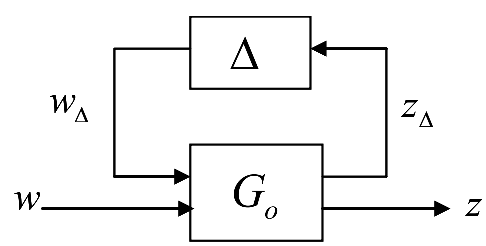



Figure **6-9**: LFT representation

In order to perform the robust control synthesis, the proposed technique requires the dimensional transformation of the bicycle model of Eq. **6.2** and Eq. **6.3**, to dimensionless representation. The transformation of the state vector $\mathbf{x} \equiv \begin{bmatrix} y & \dot{y} & \psi & \dot{\psi} \end{bmatrix}^T$ with $w = \begin{bmatrix} m & L & U \end{bmatrix}$, $u = \begin{bmatrix} length & mass & time \end{bmatrix}^T$ and time scaling $\beta = \dfrac{U}{L}$ is determined as,

$$\bar{\mathbf{x}} = \begin{bmatrix} \bar{y} \\ \bar{y}' \\ \bar{\psi} \\ \bar{\psi}' \end{bmatrix} = \mathbf{\Gamma_D}(\mathbf{x}, e, w) = \begin{bmatrix} \mathbf{\Gamma_D}(y, e, w) \\ \mathbf{\Gamma_D}(\dot{y}, e, w) \\ \mathbf{\Gamma_D}(\psi, e, w) \\ \mathbf{\Gamma_D}(\dot{\psi}, e, w) \end{bmatrix} = \begin{bmatrix} \mathbf{\Gamma_D}(y, e, w) \\ \mathbf{\Gamma_D}\left(\frac{d}{dt} y, e, w\right) \\ \mathbf{\Gamma_D}(\psi, e, w) \\ \mathbf{\Gamma_D}\left(\frac{d}{dt} \psi, e, w\right) \end{bmatrix} = \begin{bmatrix} \mathbf{\Gamma_D}(y, e, w) \\ \beta^{-1} \cdot \frac{d}{dt}(\mathbf{\Gamma_D}(y, e, w)) \\ \mathbf{\Gamma_D}(\psi, e, w) \\ \beta^{-1} \cdot \frac{d}{dt}(\mathbf{\Gamma_D}(\psi, e, w)) \end{bmatrix}$$

$$= \begin{bmatrix} \frac{1}{L} \cdot y \\ \frac{L}{U} \cdot \frac{d}{dt}\left(\frac{1}{L} \cdot y\right) \\ 1 \cdot \psi \\ \frac{L}{U} \cdot \frac{d}{dt}(\psi) \end{bmatrix} = \begin{bmatrix} \frac{1}{L} & 0 & 0 & 0 \\ 0 & \frac{1}{U} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \frac{L}{U} \end{bmatrix} \begin{bmatrix} y \\ \dot{y} \\ \psi \\ \dot{\psi} \end{bmatrix}$$

Therefore, the state transformation matrix, $\mathbf{M_x}$, is given by Eq. **6.45**.

$$\mathbf{M_x} = \begin{bmatrix} \frac{1}{L} & 0 & 0 & 0 \\ 0 & \frac{1}{U} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \frac{L}{U} \end{bmatrix} \qquad \mathbf{6.45}$$

such that, $\bar{\mathbf{x}} = \mathbf{M_x x}$. Similarly the transformation of the output vector $\mathbf{y} = [y \quad \psi]^T$ and input vector $\mathbf{u} = [\delta_f]$ are determined as,

$$\bar{\mathbf{y}} = \begin{bmatrix} \bar{y} \\ \bar{\psi} \end{bmatrix} = \mathbf{\Gamma_D}(\mathbf{y}, e, w) = \begin{bmatrix} \mathbf{\Gamma_D}(y, e, w) \\ \mathbf{\Gamma_D}(\psi, e, w) \end{bmatrix} = \begin{bmatrix} \frac{1}{L} \cdot y \\ 1 \cdot \psi \end{bmatrix} = \begin{bmatrix} \frac{1}{L} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} y \\ \psi \end{bmatrix}$$

$$\bar{\mathbf{u}} = [\bar{\delta}_f] = \mathbf{\Gamma_D}(\mathbf{u}, e, w) = [\mathbf{\Gamma_D}(\delta_f, e, w)] = [1 \cdot \delta_f] = [1][\delta_f]$$

Similarly, the output and input transformation matrices are given by Eq. **6.46** and Eq. **6.47**, respectively such that $\bar{\mathbf{y}} = \mathbf{M_x y}$ and $\bar{\mathbf{u}} = \mathbf{M_x u}$.

$$\mathbf{M_y} = \begin{bmatrix} \frac{1}{L} & 0 \\ 0 & 1 \end{bmatrix} \qquad \textbf{6.46}$$

$$\mathbf{M_u} = [1] \qquad \textbf{6.47}$$

The dimensionless system matrices are determined from the state, output, input vectors and the time scaling as defined in Eq. **6.24** and the dimensional system matrices of the bicycle model are:

$$\begin{aligned}
\bar{A} &= \beta^{-1} \cdot \mathbf{M_x} A \mathbf{M_x^{-1}} \\
&= \frac{L}{U} \cdot \begin{bmatrix} \frac{1}{L} & 0 & 0 & 0 \\ 0 & \frac{1}{U} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \frac{L}{U} \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{f_1}{mU} & \frac{f_1}{m} & -\frac{f_2}{mU} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{f_2}{I_z U} & \frac{f_2}{I_z} & -\frac{f_3}{I_z U} \end{bmatrix} \begin{bmatrix} L & 0 & 0 & 0 \\ 0 & U & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \frac{U}{L} \end{bmatrix} \\
&= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{f_1 L}{mU^2} & \frac{f_1 L}{mU^2} & -\frac{f_2}{mU^2} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{f_2 L^2}{I_z U^2} & \frac{f_2 L^2}{I_z U^2} & -\frac{f_3 L}{I_z U^2} \end{bmatrix}
\end{aligned}$$

substituting for $f_1$, $f_2$ and $f_3$ from Eq. **6.4**, yields

$$\bar{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{\left(C_{\alpha f} + C_{\alpha f}\right)L}{mU^2} & \frac{\left(C_{\alpha f} + C_{\alpha f}\right)L}{mU^2} & -\frac{a \cdot C_{\alpha f} - b \cdot C_{\alpha f}}{mU^2} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{\left(a \cdot C_{\alpha f} - b \cdot C_{\alpha f}\right)L^2}{I_z U^2} & \frac{\left(a \cdot C_{\alpha f} - b \cdot C_{\alpha f}\right)L^2}{I_z U^2} & -\frac{\left(a^2 \cdot C_{\alpha f} + b^2 \cdot C_{\alpha f}\right)L}{I_z U^2} \end{bmatrix}$$

Similarly,

$$\bar{B} = \beta^{-1} \cdot \mathbf{M_x} B \mathbf{M_u^{-1}}$$

$$= \frac{L}{U} \cdot \begin{bmatrix} \frac{1}{L} & 0 & 0 & 0 \\ 0 & \frac{1}{U} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \frac{L}{U} \end{bmatrix} \begin{bmatrix} 0 \\ \frac{C_{\alpha f}}{m} \\ 0 \\ \frac{a \cdot C_{\alpha f}}{I_z} \end{bmatrix} [1] = \begin{bmatrix} 0 \\ \frac{C_{\alpha f} L}{m U^2} \\ 0 \\ \frac{a \cdot C_{\alpha f} L^2}{I_z U^2} \end{bmatrix}$$

$$\bar{C} = \mathbf{M_y} C \mathbf{M_x^{-1}}$$

$$= \begin{bmatrix} \frac{1}{L} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} L & 0 & 0 & 0 \\ 0 & U & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \frac{U}{L} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\bar{D} = \mathbf{M_y} D \mathbf{M_u^{-1}} = \begin{bmatrix} \frac{1}{L} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} [1] = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

substituting for the numerical parameters in $\bar{A}$ and $\bar{B}$ from Table **6-2** and further expressing $\pi_2 = 1 - \pi_1$, one can simplify the above to obtain the dimensionless system matrices $\{\bar{A}, \bar{B}, \bar{C}, \bar{D}\}$ of Eq. **6.48**, in terms of the dimensionless parameters $\pi_1$, $\pi_3$, $\pi_4$ and $\pi_5$.

$$\bar{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -p_5 & p_5 & p_6 \\ 0 & 0 & 0 & 1 \\ 0 & p_4 & -p_4 & -p_3 \end{bmatrix}, \quad \bar{B} = \begin{bmatrix} 0 \\ p_2 \\ 0 \\ p_1 \end{bmatrix} \tag{6.48}$$

$$\bar{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad \bar{D} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

with $p_1$, $p_2$, …, $p_6$ as defined in Eq. **6.49**

$$\begin{aligned} p_1 &= \frac{\pi_1 \pi_3}{\pi_5} & p_4 &= \frac{\pi_4 - \pi_1(\pi_3 + \pi_4)}{\pi_5} \\ p_2 &= \pi_3 & p_5 &= \pi_3 + \pi_4 \\ p_3 &= \frac{\pi_1^2(\pi_3 + \pi_4) + \pi_4(1 - 2\pi_1)}{\pi_5} & p_6 &= \pi_4 - \pi_1(\pi_3 + \pi_4) \end{aligned} \tag{6.49}$$

In order to represent the dimensionless system in LFT form one need to separate the uncertainty from the nominal model. This is performed with the help of the block diagram representation of the model Figure **6-10**. Only parametric uncertainty is considered as the model uncertainty in this example.

It should be mentioned that this parameter-only uncertainty is artificial for this problem, and the inclusion of frequency-domain uncertainty for LFT-based controller synthesis is readily achieved by dynamic uncertainty. However, such considerations detract from this work and are discussed in previous work [22].

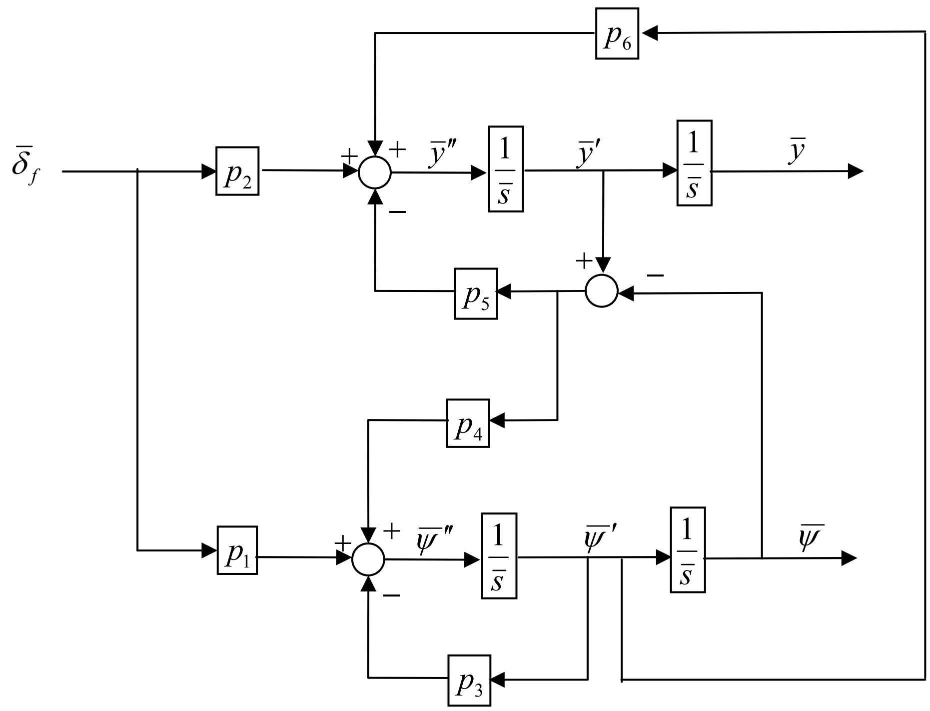



Figure **6-10**: Block diagram representation of the dimensionless bicycle model

Each parameter is expressed as $p_i = p_{io} + \delta_{pi}$, where $p_{io}$ and $\delta_{pi}$ are the nominal value and the uncertainty of the $i^{th}$ parameter, respectively. This separation is graphically shown in Figure **6-11**, where each uncertainty block $\delta_{pi}$ is replace by an input-output pair $z_i - w_i$.

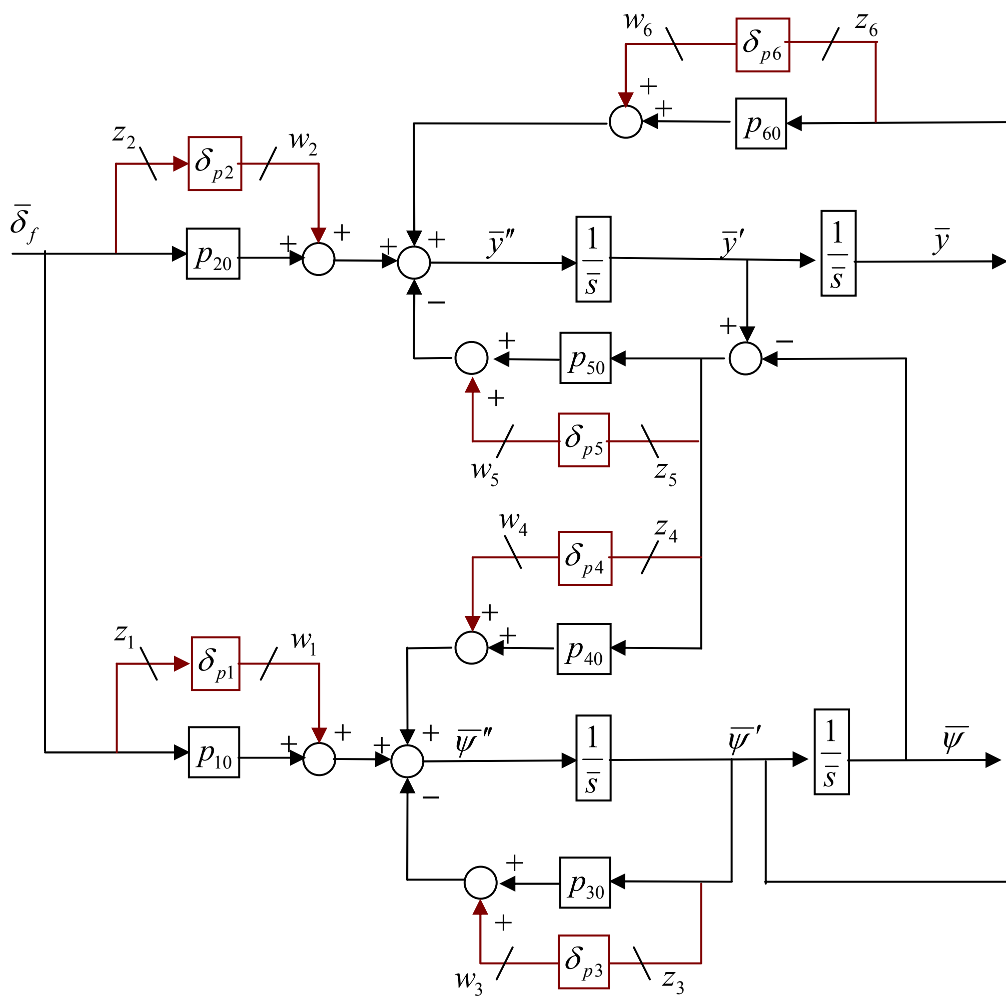



Figure **6-11**: Block diagram representation of the LFT dimensionless bicycle model

The nominal model of the system is the bicycle model at the nominal parameters namely, $p_{10}$, $p_{20}$, …, $p_{60}$, which are calculated from the nominal parameters, $\pi_{10}$, $\pi_{30}$, $\pi_{40}$ and $\pi_{50}$ using Eq. **6.49**. The nominal model is determined from Figure **6-11** and is given by Eq. **6.50**, where the $w_i$'s represent the disturbances to the system due to the parametric uncertainties.

$$\begin{aligned} \bar{y}'' &= -p_{50}\bar{y}' + p_{50}\psi + p_{60}\psi' + w_2 - w_5 + w_6 + p_{20}\delta_f \\ \bar{\psi}'' &= p_{40}\bar{y}' - p_{40}\psi - p_{30}\psi' + w_1 - w_3 + w_4 + p_{10}\delta_f \end{aligned} \qquad \textbf{6.50}$$

The state-space representation of Figure **6-11** is given by Eq. **6.51**, where the inputs and outputs are each separated into two. The two inputs are the control $\bar{\mathbf{u}}$ and the disturbance $\bar{\mathbf{w}}$ inputs while the two outputs are the output to the controller $\bar{\mathbf{y}}$ and output to the uncertainty $\bar{\mathbf{z}}$.

$$\begin{aligned}\bar{\mathbf{x}}' &= \bar{A}\bar{\mathbf{x}} + \bar{B}_1\bar{\mathbf{w}} + \bar{B}_2\bar{\mathbf{u}} \\ \bar{\mathbf{z}} &= \bar{C}_1\bar{\mathbf{x}} + \bar{D}_{11}\bar{\mathbf{w}} + \bar{D}_{12}\bar{\mathbf{u}} \\ \bar{\mathbf{y}} &= \bar{C}_2\bar{\mathbf{x}} + \bar{D}_{21}\bar{\mathbf{w}} + \bar{D}_{22}\bar{\mathbf{u}}\end{aligned} \qquad \textbf{6.51}$$

The state vector is $\bar{\mathbf{x}} \equiv \begin{bmatrix} \bar{y} & \bar{y}' & \bar{\psi} & \bar{\psi}' \end{bmatrix}^T$; the output vector to the controller is $\bar{\mathbf{y}} \equiv \begin{bmatrix} \bar{y} & \bar{\psi} \end{bmatrix}^T$ and the control input is $\bar{\mathbf{u}} \equiv \bar{\delta}_f$. The LFT structure with the state space representation given above is shown in Figure **6-12**.

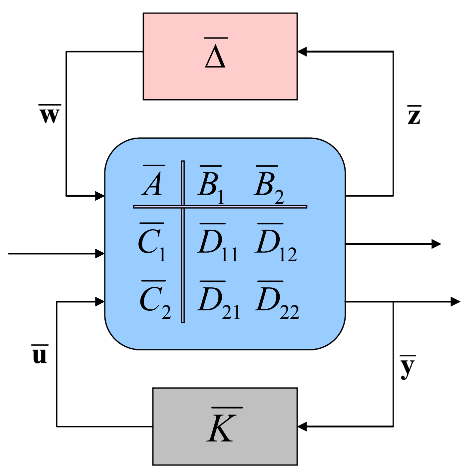



Figure **6-12**: The LFT setup of the bicycle model

For the model given in Figure **6-11**, and considering only robust stability to the variations in the parameters, the output to the uncertainty and the input disturbance vectors are $\bar{\mathbf{z}} = \begin{bmatrix} z_1 & \dots & z_6 \end{bmatrix}^T$ and $\bar{\mathbf{w}} = \begin{bmatrix} w_1 & \dots & w_6 \end{bmatrix}^T$, respectively. These vectors increase in size as performance criteria is added by outputting more signals. The system matrices of Eq. **6.51** are given by Eq. **6.52**.

$$\bar{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -p_{50} & p_{50} & p_{60} \\ 0 & 0 & 0 & 1 \\ 0 & p_{40} & -p_{40} & -p_{30} \end{bmatrix}, \bar{B}_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 1 & 0 & 0 \end{bmatrix}, \bar{B}_2 = \begin{bmatrix} 0 \\ p_{20} \\ 0 \\ p_{10} \end{bmatrix}$$

$$\bar{C}_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & -1 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \bar{D}_{11} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \bar{D}_{12} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad \textbf{6.52}$$

$$\bar{C}_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \bar{D}_{21} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \bar{D}_{22} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

The uncertainty block relates the disturbance $\bar{\mathbf{w}}$ to the outputs of the plant (inputs to the uncertainty) $\bar{\mathbf{z}}$ as, Eq. **6.53**

$$\bar{\mathbf{w}} = \bar{\Delta} \cdot \bar{\mathbf{z}} \qquad \textbf{6.53}$$

with the uncertainty block $\bar{\Delta}$ given by, Eq. **6.54**

$$\bar{\Delta} = diag\left(\begin{bmatrix} \delta_{p1} & \delta_{p2} & \delta_{p3} & \delta_{p4} & \delta_{p5} & \delta_{p6} \end{bmatrix}\right) \qquad \textbf{6.54}$$

### 6.4.2 The $\mathbf{H}_\infty$ Synthesis

The controller synthesis is performed using the $H_\infty$ control synthesis (MATLAB code given in Appendix **A.4**). The design criteria are as follows (artificially chosen):

1. Robust stability for all $\pi$ variations $|\delta_{\pi i}| \leq 0.2$. This is assumed to cover most vehicles by the distribution in Figure **6-5** and Figure **6-6**.
2. Robust performance,
   a. For an impact lateral force, the lateral displacement should be less than 0.15m/m and a settling time less than 8 sec/sec. Also the control action should have its magnitude less than 0.2 rad (~ 11.5°).
   b. For an impact yaw moment, the yaw angular displacement should be less than 0.2 (~ 11.5°) and a settling time less than 8 sec/sec. Also the control action should have its magnitude less than 0.2 rad (~ 11.5°).

The simulated response to an impulsive lateral force at the wheels of the vehicle is shown in Figure **6-13**. In this case the above requirements are met.

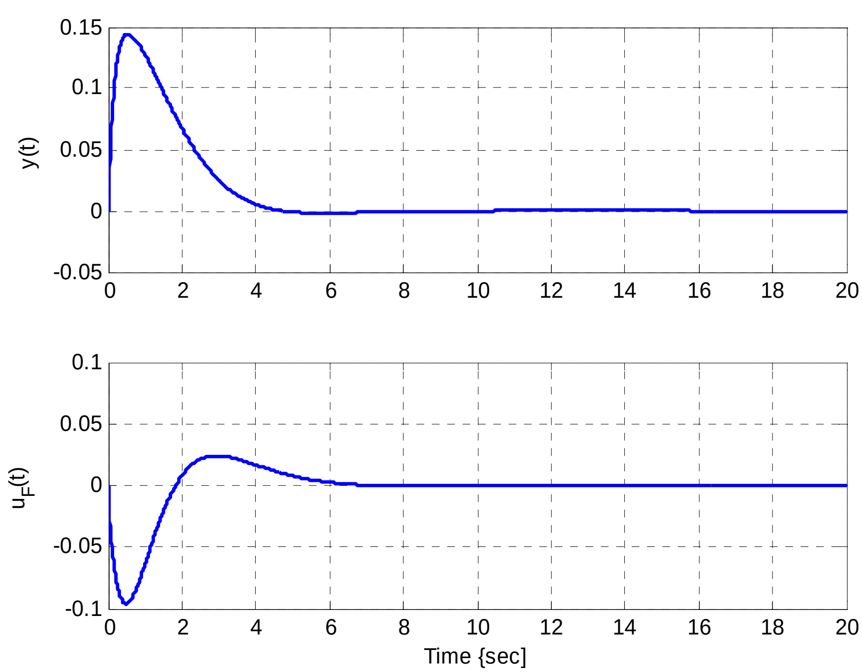



Figure **6-13**: Response to an impulsive lateral force: (top) lateral position, (bottom) control action

The response of the yaw to an impulse moment input is shown in Figure **6-14**. This also shows the performance criteria have been met.

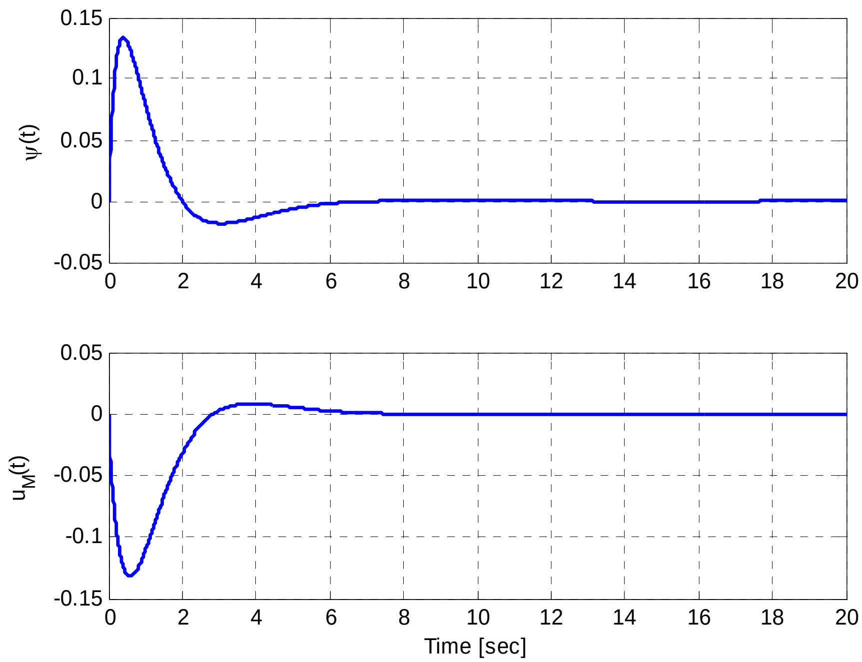



Figure **6-14**: Response to an impulse moment: (top) yaw angle, (bottom) control action

With the above being satisfactory impulse performance, the tracking of a square input, which represents sudden lane change is shown in Figure **6-15**. For the case of smooth driving (sine wave tracking), the response is shown in Figure **6-16**.

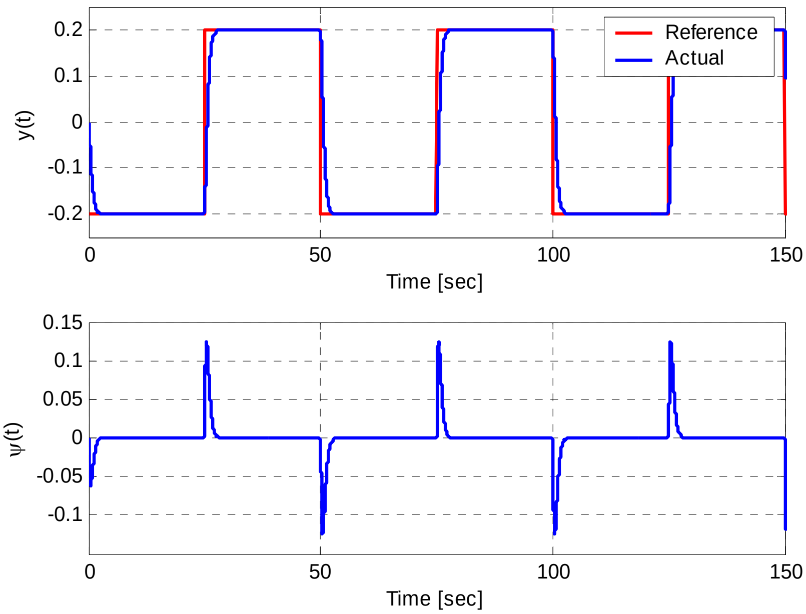



Figure **6-15**: Tracking of square path

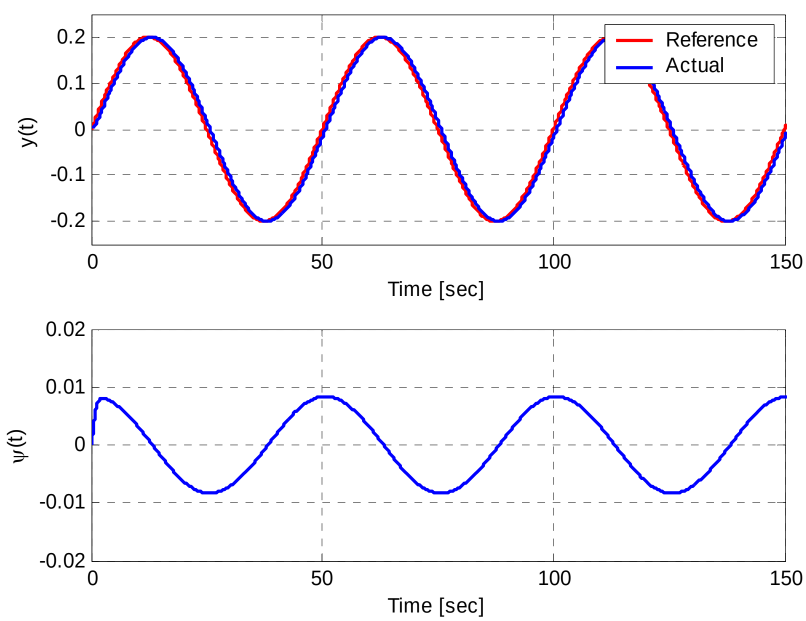



Figure **6-16**: Sine wave tracking

### 6.4.3 The $\mu$-Synthesis/Analysis

The uncertainty block extracted for the LFT model of Figure **6-11** and given by Eq. **6.54** clearly shows that it has a diagonal structure and the $H_\infty$ controller designed in the previous section is expected to be conservative as it doesn't account for the structure of the actual uncertainty. Therefore, another synthesis approached that takes into consideration the structure of the uncertainty, known as the $\mu$-synthesis is conducted to account for the structure of the uncertainty. A $\mu$-analysis was performed on the above $H_\infty$ controller to decide if $\mu$-synthesis is necessary or not. The summary of the results are given in Table **6-3** .

Table **6-3**: Summary of the $H_\infty$-synthesis / $\mu$-analysis results

| | Robust stability | Robust stability/Nominal performance |
|---|---|---|
| $H_\infty$ Norm | 0.671 | 1.012 |
| $\mu$-upper bound | 0.628 | 0.748 |
| $\mu$-lower bound | 0.627 | 0.705 |

From the results of the $\mu$-analysis, it is not required to perform $\mu$-synthesis because the values of $\mu$ and the $H_\infty$- norm are close.

## 6.5 The Experimental Implementation of the K-driver

The proposed technique is tested using a 1/5$^{th}$ scaled vehicle on a rolling roadway simulator shown in Figure **6-17**. The setup has separate board for steering and Treadmill controls and uses WinCon for real time data acquisition. The controller is implemented using SIMULINK and compiled using real time workshop with hardware target.

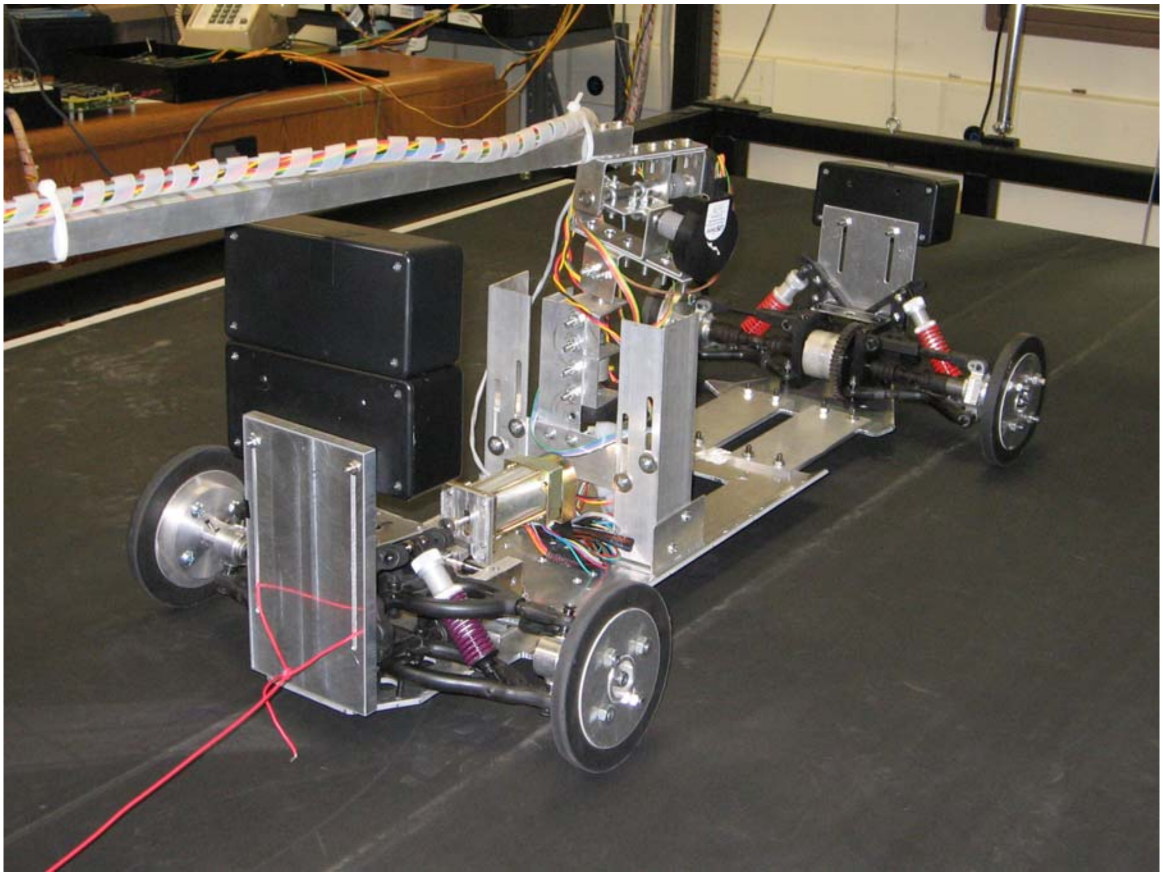


Figure **6-17**: The Scaled Vehicle and rolling roadway simulator

For the current experiment the rolling simulator is maintained horizontal. The vehicle is equipped with a steering actuator and position sensors. The position and orientation of the vehicle is measure indirectly using a linkage with encoders located at each joint. The position and orientation is calculated from the encoder readings and the dimensions of the linkage.

### 6.5.1 Actuator Dynamics and Other Practical Considerations

The steering actuator of the scaled vehicle is a motor with a gear-link mechanism and its dynamics cannot be neglected. However, it can be approximated by a second order dynamic model. The frequency response of this system is shown in Figure **6-18**.

The frequency response reveals that the system has some nonlinearity: For a linear system the frequency response is not a function of the amplitude of the input. In Figure **6-18**, however, there are slightly different frequency responses for the different input amplitudes. If such variation is significant enough to affect the overall stability of the system, then it should be included in the design of the controller design. When including this actuator dynamics in the plant model, it can be represented by a second order and a frequency dependent uncertainty.

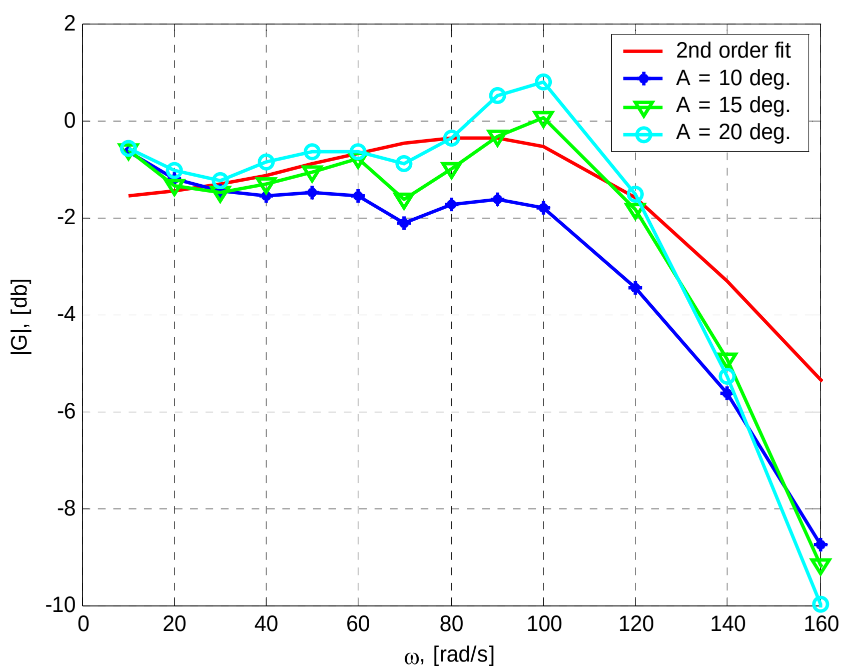



Figure **6-18**: Frequency response of the steering system

The actuator dynamics can be included in two ways: 1) augmenting the state-space model of the actuator with the plant dynamics, and 2) by representing it as frequency dependent control performance weight and including it in the controller synthesis. The second method is used in this thesis for sake of simplicity. As the control synthesis is performed in the dimensionless domain, the equivalent frequency response of the actuator dynamics in dimensionless form is determined by scaling the frequency corresponding to the time scaling. This scaling is given by Eq. **6.55**

$$\bar{\omega} = \beta^{-1}\omega \qquad \textbf{6.55}$$

where, $\beta$, $\bar{\omega}$ and $\omega$ are the time scaling, the dimensionless frequency and the dimensional frequency, respectively.

### 6.5.2 Experimental Results

The demonstration of the vehicle autopilot (K-driver) is to control the vehicle with acceptable performance under two scenarios. i) Driving under smooth lane changing and, ii) Driving under sudden lane change and maneuvering. These two scenarios are chosen to resemble real life conditions. The first scenario is normal driving where lane change is performed only to avoid slower traffic and the second scenario is chosen to mimic situations where there is an imminent danger at the current lane/position. In the case of military vehicles for example, this may represent a perceived danger of the autonomous vehicle and change of position is necessarily.

#### 6.5.2.1 Driving with Smooth Lane Changing

This operation resembles the normal driving condition on highways and city where lane-change is done merely to overtake slow vehicles on a lane. The experimental response to such driving condition is shown in Figure **6-19**.

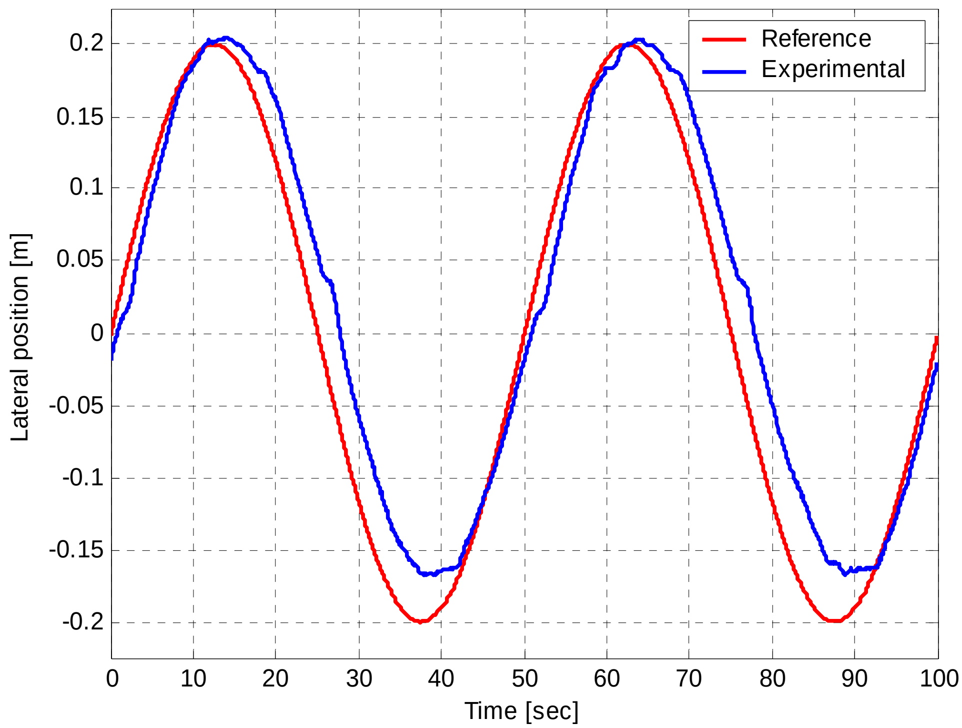



Figure **6-19**: Response to driving under smooth lane changing

#### 6.5.2.2 Driving under sudden change lane and maneuvering

This condition is to evaluate the controller performance under sudden change of lane. Sudden change of lane usually occurs in avoiding accidents that happen instantly in front of the vehicle or sudden change of mission, in the case of high performance vehicles: such as military vehicles, that required a maneuvering to suddenly change position. This scenario is an extreme case and is the most critical to evaluated the performance of the robust controller. Figure **6-20** shows the response to the driving condition under sudden change of lane.

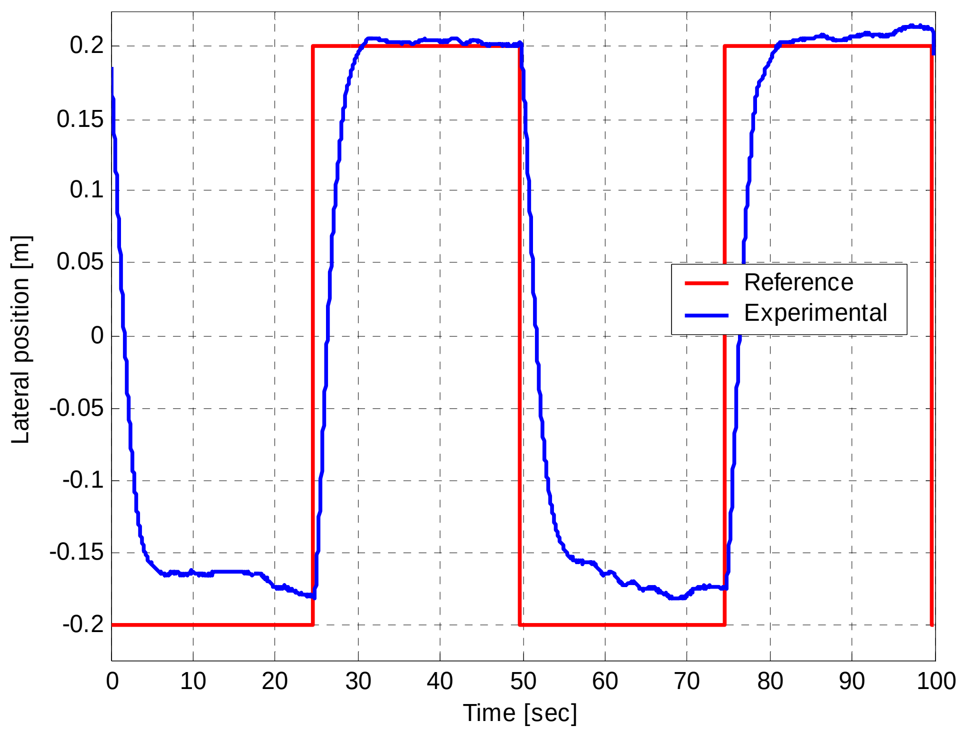
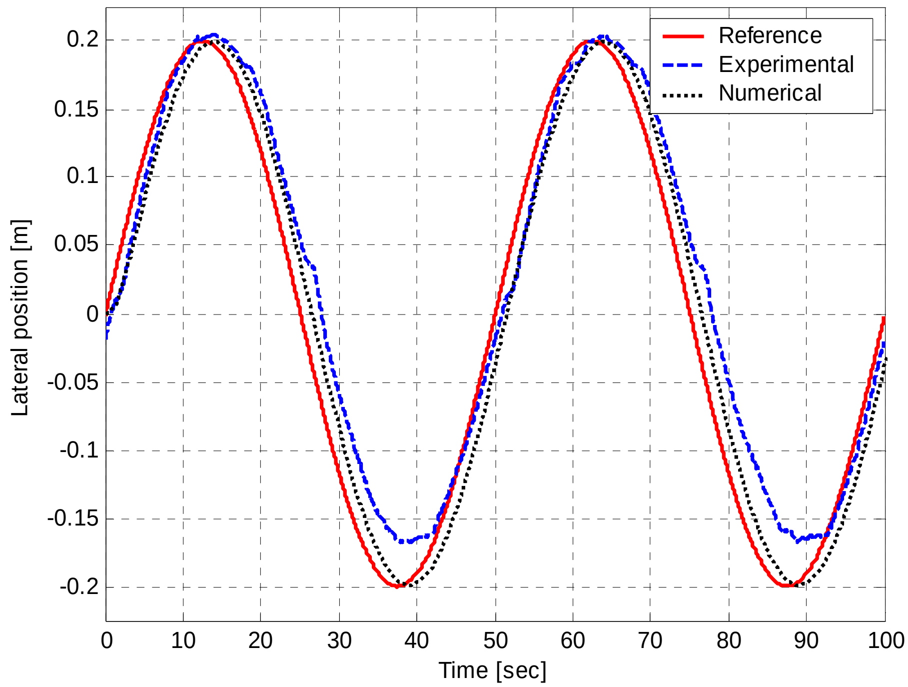
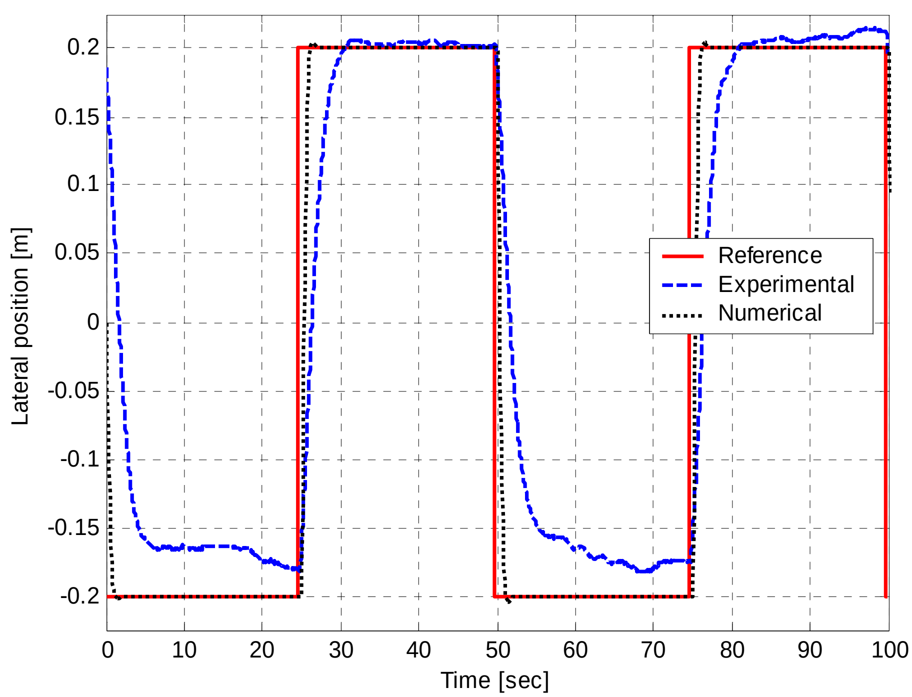



Figure **6-20**: Response to driving under sudden lane change conditions

For comparison, the experimental and simulation results are plotted in Figure **6-21**. The results show a good match. However, in this result it is observed that the experimental data are shifted up by about 7% in both the sine and square tracking plots. In other experiments this shift reached as much as 35%.




(*a*)




(*b*)

Figure **6-21**: Comparison of numerical and experimental results

The spikes in the experimental data are possibly due to some electrical problems in the measurement. These spikes are too narrow and did not affect the smooth running of the experiment.

## 6.6 Summary

The chapter focused on the development of a technique for robust control and experimental implementation using robust vehicle autopilot. The detailed process was discussed and the effectiveness of the method as an alternative approach to conventional simultaneous stabilization control was demonstrated both numerically and experimentally. In this chapter, a technique of robust control design is developed for systems that have some general coupling between the parameters. This is achieved by performing the controller synthesis in the dimensionless domain and transforming the controller back to the dimensioned domain through dimensional transformation.

## 6.7 References


[1] D. Bates and Postlethwaite, *Robust Multivariable Control of Aerospace Systems*, vol. 03. Delft: Delft University Press, 2002.

[2] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control, Design and Analysis*. Chichester, New York: John Wiley & Sons Ltd., 1996.

[3] R. A. Hyde, K. Glover, and G. T. Shanks, "VSTOL First Flight of and H Control Law," *Computing and Control Engineering Journal*, vol. 6, pp. 11-16, 1995.

[4] G. Papageorgiou, K. Glover, G. D'Mello, and Y. Patel, "Taking Robust LPV Control into Flight on the VAAC Harrier," Proceeding of the 39th IEEE Conference on Decision and Control, Sydney, Australia, 2000.

[5] G. J. Balas and J. C. Doyle, "Robustness and performance trade-offs in control design for flexible structures," *Control Systems Technology, IEEE Transactions on*, vol. 2, pp. 352-361, 1994.

[6] H. Buschek, "Full Envelope Missile Autopilot Design Using Gain Scheduled Robust Control," *AIAA Journal of Gauidance, Control and Dynamics*, vol. 22, pp. 115-122, 1999.

[7] G. Ferreres and M. M'Saad, "Parametric Robustness Evaluation of a H_inf Missile Autopilot," *AIAA Journal of Gauidance, Control and Dynamics*, vol. 19, pp. 621-627, 1996.

[8] I. Fialho and G. J. Balas, "Gain-Scheduled Lateral Control of the F-14 Aircraft During Powered Approach Landing," *AIAA Journal of Gauidance, Control and Dynamics*, vol. 23, pp. 450-459, 2000.

[9] R. J. Niewoehner and I. I. Kaminer, "Design of an Autoland Controller for an F-14 Aircraft Using H-inf Synthesis," *AIAA Journal of Gauidance, Control and Dynamics*, vol. 19, pp. 656-663, 1996.

[10] E. A. Jonckheere and Y. Gwo-Ruey, "Propulsion control of crippled aircraft by H_inf model matching," *Control Systems Technology, IEEE Transactions on*, vol. 7, pp. 142 - 159, 1999.

[11] Y. Ochi and K. Kanai, "Automatic approach and landing for propulsion controlled aircraft by H<sub>∞</sub> control," Proceedings of the IEEE Conference on Control Applications, Kohala Coast-Island of Hawaii, Hawaii, 1999.

[12] N. Sivashankar, I. Kaminer, and D. Kuechenmeister, "Design, analysis and hardware-in-the-loop simulation of a MIMO controller for a VTOL unmanned aerial vehicle using H_inf synthesis," Proceedings of the American Control Conference, Baltimore, MD, 1994.

[13] D. H. Baldelli, H. Matsushita, M. Hashidate, and K. Saitoh, "Flutter Margin Augmentation Synthesis Using Normalized Coprime Factors Approach," *AIAA Journal of Gauidance, Control and Dynamics*, vol. 18, pp. 802-811, 1995.

[14] D. G. Bates, S. L. Gatley, and I. Postlethwaite, "Design and Piloted Simulation of a Robust Integrated Flight and Propulsion Controller," *AIAA Journal of Gauidance, Control and Dynamics*, vol. 23, pp. 269-278, 2000.

[15] J. P. Friang, G. Duc, and J. P. Bonnet, "Robust Autopilot for and Air to Ground Missile: Loop Shaping H_infinity Design, Gap-Metric and Real Mu-Analysis," Proceedings of the AIAA Conference on Guidance, Navigation and Control, 1996.

[16] J. Wang and M. Tomizuka, "Robust Hinfinty Lateral Control of Heavy-duty Vehicles in the Automated Highway System," American Control Conference, San Diego, CA, USA, 1999.

[17] M. Tai and M. Tomizuka, "Robust Lateral Control of Heavy Duty Vehicles: Final Report," University of California, Berkeley July 2003.

[18] J. Wang and M. Tomizuka, "Gain-scheduled H_inf Loop-shaping Controller for Automated Guidance of Tractor-Semitrailer Combination Vehicles," American Control Conference, Chicago, IL, USA, 2000.

[19] P. Hingwe, A. Packard, and M. Tomizuka, "LPV Controller for Lateral Control of Heavy Vehicles," American Control Conference, Chicago, IL, 2000, 2000.

[20] B. A. Guvenc, T. Bunte, D. Odenthal, and L. Guvenc, "Robust two degree-of-freedom vehicle steering controller design," *Control Systems Technology, IEEE Transactions on*, vol. 12, pp. 627, 2004.

[21] M. Hosaka and T. Murakami, "Yaw rate control of electric vehicle using steer-by-wire system," The 8th IEEE International Workshop on Advanced Motion Control, Kawasaki, Japan, 2004.

[22] S. N. Brennan, "On Size and Control: The Use of Dimensional Analysis in Controller Design," in *Mechanical Engineering*. Urbana-Champaign: University of Illinois, 2002, pp. 370.

[23] J. Ackermann, "Robust control prevents car skidding," *Control Systems Magazine, IEEE*, vol. 17, pp. 23-31, 1997.

[24] J. Sun, A. W. Olbrot, and M. P. Polis, "Robust stabilization and robust performance using model reference control and modeling error compensation," *Automatic Control, IEEE Transactions on*, vol. 39, pp. 630-635, 1994.

[25] S. Brennan and A. Alleyne, "Generalized H-infinity Vehicle Control Utilizing Dimensional Analysis," American Control Conference, Denver, CO, 2003.

[26] S. Brennan and A. Alleyne, "The Illinois Roadway Simulator: a mechatronic testbed for vehicle dynamics and control," *Mechatronics, IEEE/ASME Transactions on*, vol. 5, pp. 349-359, 2000.

[27] A. Alleyne, "A Comparison of Alternative Obstacle Avoidance Strategies for Vehicle Control," *Vehicle System Dynamics*, vol. 27, pp. 371-392, 1997.

[28] W. Sienel, "Estimation of the tire cornering stiffness and its application to active car steering," 1997.

[29] M. Arndt, E. L. Ding, and T. Massel, "Identification of cornering stiffness during lane change maneuvers," 2004.

# Chapter 7

# Summary and Future Work

## 7.1 Introduction

The transformation of system representations from dimensioned to dimensionless forms has been shown to be a very effective means of simplifying analysis and design of control systems. Because of this transformation, more complex systems can now be analyzed and solutions can be sought much easier, more robustly, and/or more generalized than a design based on the original system representation.

The reduction in parameters has a profound impact on the performance of gain scheduling methods because any decrease in the dimension of the parametric space implies significant (exponentially smaller) problem simplification and controller representation.

Similarly, the reduction in size of the parametric uncertainty block has a significant advantage on the synthesis of robust controllers, specifically in reducing computational complexity and conservativeness of the control system. The size of the parametric uncertainty block imposes strong limitations on many robust control design tools, specifically the $\mu$-synthesis designs that make use of additional information such as the structure of the uncertainty block to reduce conservatism. For example, in the $\mu$-synthesis framework, an exact value of $\mu$ can be computed only for uncertainty block size of three or less. For uncertainty block size greater than three, only the range of $\mu$

can be evaluated and this range can be arbitrarily large. In this case control synthesis based on the upper value of $\mu$ is used and is hence more conservative.

The development of adaptive systems as an open/interactive system to adapt to known changes in parameters have the advantage of developing agile systems that can control a family of systems. One such example is the vehicle autopilot system developed and implemented in this research.

## 7.2 Summary

Novel methods were presented that relieve the limitations of current theories and practical design methods of gain scheduling and robust controllers. The method is called dimensional transformation and is based on the re-parameterization of the system description in an equivalent form, namely dimensionless representation of the system. This is generally performed through the scaling of the signals or states and parameters by other parameters that suits the intended advantage.

### 7.2.1 Gain Scheduling and Parametric Space Minimization

It was shown that, by representing dynamic systems using dimensionless description, the gain scheduling parameters can be reduced significantly by a factor of up to an order magnitude. The dimensional transformation matrices, i.e., choice of scaling parameters, are not unique and higher degree of gain-scheduling parameter reduction is achieved by careful choice of the scaling and scaled parameters during the dimensional transformation. Therefore, for a given set of transformation matrices, a mathematical rule to quantify number of scheduling parameters in the new representation was formalized and presented as a theorem. It was also shown that under special cases, an LPV system may be converted to an LTI system during such transformation.

Equivalence between dimensionless and dimensioned controllers was demonstrated using a gantry system. The gantry results illustrate that, by carefully choosing certain gain-scheduling variables for dimensional transformations, gain-scheduling parameters can be reduced by mapping the system to a lower parameter space (i.e., from three to two, in the case of the gantry system) in the non-dimensional system representation.

The problem of finding the optimum transformation for gain scheduling – e.g. one that minimizes the necessary number of gain-scheduling parameters – was solved using a combinatorial optimization algorithm. The optimization is required because the dimensionless representation of a given dimensional system is non-unique. It is also non-trivial to find the best system representation that yields the minimum number of gain scheduling parameters.

### 7.2.2 Robust Control and Parametric block reduction

It was shown in the general structured uncertainty case that a lower $H_\infty$- norm is achieved by using dimensional transformation to reduce the size of the uncertainty block in the system description. This lower norm increases the allowable perturbation of the system in the new representation. Also, better performance in the form of reduced overshot and control effort was demonstrated using the dimensionless representation compared to the dimensional representation.

Using the return difference matrix determinant condition for robust stability, it was also shown more clearly that the dimensionless representation will give a higher allowable parametric perturbation than the dimensional form.

Finally three main practical limitations of the method were identified and potential remedies established. The first limitation is that the dimensionless uncertainty description may have wider uncertainty bound than the dimensional case. While this may not usually outweigh the merits of the technique, it may be possible in some rare cases it have a larger impact on the effectiveness of the current technique. The second limitation is due to input/output uncertainty scaling. When the input/output scaling uses uncertain parameters, then the effectiveness of the controller to robustly control the system may be compromised. However, this can be mitigated by re-inclusion of uncertainty into the system representation to account for uncertain scaling factors. This is done at the expense of reduction of the uncertainty block size. The last drawback identified in this work is the uncertainty in time scaling. Similar to the input/output scaling, this arises from the scaling of time by uncertain parameters whose only nominal values are known. For the uncertainty time scaling case, a thorough check is recommended to confirm the robustness of the controller under uncertain model or signal scaling. This was shown using the mass-spring-damper example where in this specific example, the method still proved to be more effective than the classical system representation.

### 7.2.3 The Vehicle Autopilot: A Simultaneous Robust Control through Parametric Adaptation

A robust simultaneous control and parametric adaptation technique developed in this thesis was shown to be effective for systems with inherent parametric couplings. Dimensional transformation for dynamic controllers was presented for the purpose of parametric adaptation. A test of the method on a scale-sized rolling-roadway simulator system showed the feasibility of the developed technique. The effectiveness of the method as an alternative approach to conventional simultaneous stabilization control was demonstrated both numerically and experimentally.

The two potential applications of this technique are: *1. Simultaneous control through parametric adaptation* and *2. Prototyping and testing of similar systems.*

*1. Simultaneous control through parametric adaptation:* Control synthesis for autonomous and high performance vehicles (such as military vehicles) usually requires the consideration of many operating regimes. Depending on the complexity of the mission, the controller is required to perform to the expected level. Whenever, the mission requires the coordination of different sizes of vehicles, one needs to have different controller designs for the different vehicles. The current approach helps develop a technique to simultaneously control all systems in the mission through parametric adaptation. This approach also benefits manufacturers of passenger vehicles to develop a single controller for different vehicle models in their production line by carefully choosing the nominal vehicle for the controller design.

*2. Prototyping and testing of similar systems:* The technique proposed here can be used in experimental prototyping of similar systems Figure **7-1**, i.e., systems that exhibit the same dimensionless parameters. For example: studying autonomous-vehicle-following-control at a smaller scale. The results can be scaled in the same way an airfoil in a wind tunnel is scaled to study the performances of an airplane.


Figure **7-1**: Scaling of systems

### 7.3 Future Work

The results of the current work encourage the extension of these techniques to new problem areas. The author of this thesis has identified the following as areas of potential future research in this line of work. These include: 1) Development of the technique for non-parametric, frequency-dependent uncertainty size and order reduction, 2) Development of a technique for design of variable band filters, 3) Implementation of the vehicle autopilot to full size vehicles, 4) Implementation of the developed technique on a variable clock processor. The first two are areas that require extensive research and the last two involve implementation of the tools developed in this thesis to systems of larger scale both dimensional and time scales.

### 7.3.1 Development of a Technique for Non-parametric, Frequency Dependent Uncertainty Size and Order Reduction

The implementation of dimensional transformation can have an impact on the order and size reduction of the uncertainty block. Frequency dependent uncertainty is generally non-parametric and usually the result of neglected higher order dynamics of the system. An extension of the technique discussed in this thesis to such uncertainty models can potentially reduce the order of the uncertainty block resulting in a lower-order controller. For example the product of two frequency dependent uncertainty functions may results in a new uncertainty description that has lower order function than either of the uncertainties. In the ideal case where the two uncertainty functions are an inverse to each other, the result is a constant which is a zero order system. This has a significant effect in reducing the controller order in the robust controller synthesis.

### 7.3.2 Development of a Technique for Design of Variable Band Filters

For systems whose noise-frequency increases with an increase in the excitation or interrogation frequency of the system, it is important to have a filter that continuously adapts to the change to insure effective separation of the actual signal from the noise. A filter with fixed cutoff frequencies may be effective in some applications. In such application the dimensional transformation method can be used to design a filter in the dimensionless domain and parametrically adapts based on the known systems frequency. In other words, this method of filtering would represent a dimensional scaling of system dynamics with respect to time-variable stochastic measures rather than deterministic parameters.

### 7.3.3 Implementation of the Vehicle Autopilot to Full Scale Vehicles

The vehicle autopilot developed in this thesis was implemented on the scaled vehicle. Future work in this line can focus on matching the performance of the scaled vehicle to full size vehicle and study the vehicle behavior.

### 7.3.4 Implementation of the Current Techniques on Variable Clock Processors

The robust and gain scheduling controllers in this thesis were designed in dimensionless domain under changing and time-dependent time scales. Expanding the current research concept of time scaling, especially translating it into a hardware level is an important area of research. Further, analyzing dynamic models in this time scale and studying their interaction with systems running in real time. The studies at this time scale, if proved successful, can lead to the realization of the current and future research in the dimensionless systems representation into a commercial level dimensionless system implementation that benefits the society at large.

# Appendix A

# Computer Code

In this section the different computer codes developed are listed along with all input and output data. In the listing of the codes, first the main program is listed followed by the functions in the same order they are called by the main program or the subsequent functions.

## A.1 The method of transformation process

The code for the dimensional transformation process is developed based on the combinatorial algorithm of searching for the scaling transformation matrix that minimizes the number of gain-scheduling parameters. Since there are only finite numbers of possible combinations, the algorithm searches through all and applies the theorem to find the reduction associated with the current. It finally returns all the combinations that have the optimum solution.

```
%=========================================
% The Brennan Research Group, July 2005
% Haftay Hailu, Ph.D Student
% Sean Brennan (Asst. Prof. ), Advisor
%
% Functions to be included:
%    1. V_Complement.m
%    2. Compute_Cs.m
%    3. isIn.m
%========================================
clear all ;
% The following are examples of the

% Define all variable symbols
```

```
syms t x theta u b g mt mp L ;
Ne = 3 ;    % size of basis unit vectors

S_Name = [t x theta u] ;
S_Du = [0 0 1; 1 0 0; 0 0 0; 1 1 -2]' ; % S = signals
Q_Name = [b g] ;
Q_Du = [0 1 -1; 1 0 -2]' ;        % Q = constant parameters
P_Name = [mt mp L] ;
P_Du = [0 1 0; 0 1 0; 1 0 0]' ;   % P = varrying parameters

QP_Name = [Q_Name P_Name] ;
[AD, BD, I_AD, I_BD, N_opt, Err] = Optimum_AD(S_Du, Q_Du, P_Du, Ne) ;
for i = 1:N_opt
    i
    Cs = Compute_Cs(AD(:,:,i),BD(:,:,i)) ;
    BD_Name = [S_Name QP_Name(I_BD(i,:))] ;
    AD_Name = QP_Name(I_AD(i,:)) ;
    [Pii, Err] = Generate_Pi(BD_Name, AD_Name, Cs) ;
    disp(Pii)
end


function [AD, BD, I_AD, I_BD, N_opt, Err] = Optimum_AD(S_Du, Q_Du,
P_Du, Ne) ;

%=========================================
% The Brennan Research Group, July 2005

% Functions to be included:
%    1. V_Complement.m
%    2. Compute_Cs.m
%    3. isIn.m
%=========================================
R_pi_max = 0 ;
R_pi = 0 ;
N_opt = 0 ;
I_AD = [] ;
I_BD = [] ;
[Nu, Nq] = size(Q_Du) ;
QP_Du = [Q_Du P_Du] ;
[Nu, NqNp] = size(QP_Du) ;
if Nu ~= Ne
    disp('=======================================================') ;
    disp('Error: Size mismatch between the unit basis dimensions ') ;
    disp('       and the unit dimensional vector                 ') ;
    disp('=======================================================') ;
    Err = 1 ; AD = [] ; BD = [] ; N_opt = 0 ;
    return
else
    Err = 0 ;
    NCi = nchoosek(1:NqNp,Nu)
    [N_ADi, Nu] = size(NCi)
end
for i = 1:N_ADi
    Np2 = 0 ;
```

```
    for j = 1:Nu
        ADi(:,j) = QP_Du(:,NCi(i,j)) ;
        if NCi(i,j) > Nq
            Np2 = Np2 + 1 ;
        end
    end
    [NC_C(i,:), Err1] = V_Complement(NCi(i,:), 1:NqNp) ;
    if cond(ADi) < 1000     % check if AD is singular
        k = 0 ;
        BD_q = [] ;
        for j = 1:NqNp-Nu
            BDi_QP(:,j) = QP_Du(:,NC_C(i,j)) ;
            if NC_C(i,j) <= Nq
                k = k + 1 ;
                BD_q(:,k) = QP_Du(:,NC_C(i,j)) ;
            end
        end
        if isempty(BD_q) == 1
            R_pi = Np2
        else
            Cs_q = -(inv(ADi)*BD_q)' ;
            Cs_q1p2 = Cs_q(:,Nu-Np2+1:Nu) ;
            R_pi = Np2 - rank(Cs_q1p2)
        end
        if R_pi > R_pi_max
            N_opt = 1
            R_pi_max = R_pi ;
            AD = [] ;
            BD = [] ;
            I_AD = [] ;
            I_BD = [] ;
            AD(:,:,N_opt) = ADi ;
            BD(:,:,N_opt) = [S_Du BDi_QP] ;
            I_AD(N_opt,:) = NCi(i,:) ;
            I_BD(N_opt,:) = NC_C(i,:) ;
        elseif R_pi == R_pi_max
            N_opt = N_opt + 1
            AD(:,:,N_opt) = ADi ;
            BD(:,:,N_opt) = [S_Du BDi_QP] ;
            I_AD(N_opt,:) = NCi(i,:) ;
            I_BD(N_opt,:) = NC_C(i,:) ;
        end
    end
end


function [Cs, Err] = Compute_Cs(A,B) ;
%==========================================
% The Brennan Research Group, July 2005
%
%=========================================

if cond(A) > 1000
    Err = 1 ;
    Cs = [] ;
```

```
    return
else
    Err = 0 ;
    Cs = (-inv(A)*B)' ;
end


function [Pii, Err] = Generate_Pi(BD_Name, AD_Name, Cs) ;


%==========================================
% The Brennan Research Group, July 2005

% Functions to be included:
%    1. V_Complement.m
%    2. Compute_Cs.m
%    3. isIn.m
%=========================================

[r, c] = size(AD_Name) ;
[r1, c1] = size(BD_Name) ;
[r2, c2] = size(Cs) ;

if ((c1 ~= r2)||(c ~= c2))
    Err = 1 ;
    return
else
    Err = 0 ;
    for i = 1:c1
        temp = 1 ;
        for j = 1:c2
            temp = temp*AD_Name(j)^Cs(i,j) ;
        end
        Pii(i,1) = BD_Name(i)*temp ;
    end
end


function [Vc, Err] = V_Complement(V,U) ;

%==========================================
% The Brennan Research Group, July 2005

% Functions to be included:
%    1. isIn.m
%
%=========================================

[r1,c1] = size(U) ;
[r2,c2] = size(V) ;
if (((r1 > 1)&&(c1 > 1))||((r2 > 1)&&(c2 > 1)))
    disp('==================================') ;
    disp('Error: Arguements must be vectors ') ;
    disp('==================================') ;
    Err = 1 ;
```

```
    Vc = [] ;
    return
else
    Err = 0 ;
    k1 = 0 ;
    L1 = length(U) ;
    for i = 1:L1
        k2 = isIn(U(i),V) ;
        if k2 == 0
            k1 = k1 + 1 ;
            Vc(k1) = U(i) ;
        end
    end
end

function [kk, Err] = isIn(x,U) ;
%=========================================
% The Brennan Research Group, July 2005
%
%========================================
[r1,c1] = size(U) ;
[r2,c2] = size(x) ;
if (((r2 > 1)||(c2 > 1))||((r1 > 1)&&(c1 > 1)))
    disp('==========================================') ;
    disp('Error: The first arguement must be scalar ') ;
    disp('       and the second arguement a vector ') ;
    disp('==========================================') ;
    Err = 1 ;
    kk = 0 ;
    return
else
    L1 = length(U) ;
    Err = 0 ;
    kk = 0 ;
    for i = 1:L1
        if x == U(i)
            kk = kk + 1 ;
        end
    end
end
```

### A.2 Code for the ACC Benchmark Problem

```
%===========================================================%
% The Brennan Research Group, March 2006                    %
% Haftay Hailu, Ph.D Student                                %
% Sean Brennan, Advisor                                     %
%                                                           %
% The ACC Benchmark Problem - DIMENSIONAL DOMAIN            %
```

```
% Controller design for the ACC Benchmark Problem            %
% Design Specs:                                              %
%   I. Robust stablity for all values of |delta_i| <= 0.2    %
%  II. For the nominal plant                                 %
%        1. |u| <= 1                                         %
%        2. Settling time ~ 15 sec (on nominal plant)        %
%                                                            %
% User functions to be inlcluded, when running this code     %
%   1. minsys                                                %
%------------------------------------------------------------*

% First create the generalized plant. Inputs w_k,w_m1,w_m2, w, u
%                                     Outputs z_k, z_m1, z_m2, y

clear all ;     close all ; clc ;
A = [0 1 0 0; -1 0 1 0; 0 0 0 1; 1 0 -1 0] ;
B1 = [ 0 0 0 0; -1 -1 0 0; 0 0 0 0; 1 0 -1 1] ;     % Inputs w_k, w_m1,
w_m2 and w
B2 = [0 1 0 0]' ;                                    % Input Controller,
u
C1 = [1 0 -1 0; -1 0 1 0; 1 0 -1 0; 0 0 1 0] ;       % Outputs z_k,
z_m1, z_m2 and y for performance spcs
C2 = [0 0 1 0] ;                                      % output to the
controller
D11 = [0 0 0 0; -1 -1 0 0; 1 0 -1 1; 0 0 0 0] ;      % No Inputs are
Output at this time
D12 = [0 1 0 0]' ;  D21 = [0 0 0 0] ;   D22 = 0 ;
B = [B1 B2] ;   C = [C1; C2] ;  D = [D11 D12; D21 D22] ;
P = pck(A, B, C, D) ;

% 1. Designing a Controller for Robust Stability

% == Transform the plant via a 'pole-shifting' bilinear transform:
p1 = -0.34 ;                                        % p1 control settling
time;
p2 = -300 ;                                        % p2 controls bandwidth
[Ab,Bb,Cb,Db] = bilin(A,B, C, D,1,'Sft_jw',[p1 p2]);

% == H-Inf Design:
Pb = pck(Ab,Bb,Cb,Db) ;
[k,g] = hinfsyn(Pb,1,1,0,100,1e-2);

% == Transform the controller back to the original domain
[Akb,Bkb,Ckb,Dkb] = unpck(k) ;
[Ak,Bk,Ck,Dk] = bilin(Akb,Bkb,Ckb,Dkb,-1,'Sft_jw',[p1 p2]) ;
K = pck(Ak,Bk,Ck,Dk) ;

% == Check for the closed loop poles
Pcl = starp(P,K);
spoles(Pcl)

% == Check the hinfty norm, if it's " < 5 " --> robust stability
hinfnorm(sel(Pcl,(1:3),(1:3)))      %....>5 --> Robust stability failed
pause
```

```
% 2. Designing a Controller for Robust Stability and Performance

% == Check if Robust Performance is satisfied as it is
% ==== a. Settling time  due to the impulse disturbance
T = [0:0.01:20] ;
subplot(211) ;
[acl,bcl,ccl,dcl] = unpck(sel(Pcl,4,4));
sys = ss(acl,bcl,ccl,dcl) ;
[y,T] = impulse(sys,T);
plot(T,y); axis auto ; ylabel('x_2(t)') ;   grid ;


% ==== b. Control action  due to the impulse disturbance
% Need to modify the plant to have the displacement and control
% action available as performance outputs.
C1 = [C1;0 0 0 0];    C = [C1; C2] ;
D11 = [D11;0 0 0 0] ;  D12 = [D12;1] ; % Output u for performances spcs
D = [D11 D12; D21 D22] ;
Subplot(212) ;
P = pck(A, B, C, D);
Pcl = starp(P,K);
[acl,bcl,ccl,dcl] = unpck(sel(Pcl,5,4));
sys = ss(acl,bcl,ccl,dcl) ;
[u,T] = impulse(sys,T);             % The control action is large!!
plot(T,u); axis auto ; ylabel('u(t)') ; grid ;
pause
%
% Need to put a weight in the control action and displacement
% To guess the parameters of these filters look at the frequency
response
close all ;
subplot(211)
omega = logspace(-2,4);
out_u = frsp(sel(Pcl,5,4), omega); % control freq response
vplot('liv,lm',out_u)
subplot(212)
out_x2 = frsp(sel(Pcl,4,4), omega); % displacement freq response
vplot('liv,lm',out_x2)
pause

% select cutoff freq for the control action
%   Note, using a second order weight to get sharper cut-off
Wu = nd2sys(conv([1 .8],[1 .8]),conv([1,50],[1,50]),7000);

% select cutoff freq for the displacement also at about 1 Hz
Wz = nd2sys([1 1],[1 1],1e-3);
%
% Plot the weights and the actual frequency responses
%
close all ;
outwx = frsp(Wz,omega); % output weight frequency response
outwu = frsp(Wu,omega); % control weight frsp
subplot(121)
vplot('liv,lm',outwx,out_x2,outwu,out_u)
legend('output weight','X_2(\omega)','control weight','U(\omega)')
```

```
pause
%
% Add a little bit of uncertainty weight to
% force the Hinfty norm to be less than 5
%
Wd = [0.2 0.2 0.2];
W = daug(Wd,Wz,Wu,1);
Pa = mmult(W,P);
Pa = minsys(Pa);
[Aa,Ba,Ca,Da] = unpck(Pa);

%       2.1 Hinfinty Controller Design
% == Transform the plant via a 'pole-shifting' bilinear transform:
[Ab,Bb,Cb,Db]=bilin(Aa,Ba,Ca,Da,1,'Sft_jw',[p1 p2]);

% == H-Inf Design:
Pb = pck(Ab,Bb,Cb,Db);
[k,g] = hinfsyn(Pb,1,1,0,1e10,1e-2);

% == Transform the controller back to the original domain
[Akb,Bkb,Ckb,Dkb] = unpck(k);
[Ak,Bk,Ck,Dk] = bilin(Akb,Bkb,Ckb,Dkb,-1,'Sft_jw',[p1 p2]);
K = pck(Ak,Bk,Ck,Dk);
minfo(K)
% == Check for the closed loop poles
Pcl = starp(P,K);
spoles(Pcl)

% == Check the hinfty norm, if it's < 5 --> robust stability
M = sel(Pcl,(1:3),(1:3)) ;
hinfnorm(M)

% now check the settling time and control action due to the impulse
disturbance
%figure(2)
subplot(222)
[acl,bcl,ccl,dcl]=unpck(sel(Pcl,4,4));  % Displacement
sys = ss(acl,bcl,ccl,dcl) ;
[y,T] = impulse(sys,T);
plot(T,y); axis auto ; ylabel('x_2(t)') ;   grid ;

subplot(224)
[acl,bcl,ccl,dcl]=unpck(sel(Pcl,5,4));  % Control Action
impulse(acl,bcl,ccl,dcl);
sys = ss(acl,bcl,ccl,dcl) ;
[u,T] = impulse(sys,T);
plot(T,u); axis auto ; ylabel('u(t)') ;     grid ;


%============================================================%
% The Brennan Research Group, March 2006                     %
% Haftay Hailu, Ph.D Student                                 %
% Sean Brennan, Advisor                                      %
%                                                            %
% The ACC Benchmark Problem - DIMENSIONLESS DOMAIN           %
```

```
% Controller design for the ACC Benchmark Problem          %
% Design Specs:                                            %
%   I. Robust stablity for all values of |delta_i| <= 0.2  %
%  II. For the nominal plant                               %
%        1. |u| <= 1                                       %
%        2. Settling time ~ 15 sec (on nominal plant)      %
%                                                          %
% User functions to be inlcluded, when running this code   %
%   1. minsys                                              %
%----------------------------------------------------------*

% First create the generalized plant. Inputs w_Pi2, w, u
      %                                    Outputs zPi2, y

clear all ;     close all ; clc ;

% Nominal Parameters and corresponding uncertainities
m1o = 1.0 ;    m2o = 1.0 ;    ko = 1.0 ;
delta_m1 = 0.2 ; delta_m2 = 0.2 ; delta_k = 0.2 ;

% Nominal Dimensionless Parameters and corresponding uncertainities
Pi2o = m2o/m1o ;
Pi2min = (m2o - delta_m2)/(m1o + delta_m1) ;
Pi2max = (m2o + delta_m2)/(m1o - delta_m1) ;
delta_Pi2 = max(Pi2max - Pi2o , Pi2o - Pi2min) ;

A = [0 1 0 0; -1 0 1 0; 0 0 0 1; 1/Pi2o 0 -1/Pi2o 0]
B1 = [ 0 0; 0 0; 0 0; -1/Pi2o 1/Pi2o] ;     % Inputs w_Pi2, w
B2 = [0 1 0 0]' ;                           % Input Controller, u
C1 = [1/Pi2o 0 -1/Pi2o 0; 0 0 1 0] ;        % Outputs zPi2, y
C2 = [0 0 1 0] ;                            % output to the controller
D11 = [-1/Pi2o 1/Pi2o; 0 0] ;                 % No Inputs are Output at
this time
D12 = [0 0]' ;  D21 = [0 0] ;   D22 = 0 ;
B = [B1 B2] ;   C = [C1; C2] ;  D = [D11 D12; D21 D22] ;
P = pck(A, B, C, D) ;

% 1. Designing a Controller for Robust Stability

% == Transform the plant via a 'pole-shifting' bilinear transform:
p1 = -0.3 ;                                       % p1 control settling
time;
p2 = -1000;                                    % p2 controls bandwidth
[Ab,Bb,Cb,Db] = bilin(A,B, C, D,1,'Sft_jw',[p1 p2]);

% == H-Inf Design:
Pb = pck(Ab,Bb,Cb,Db) ;
[k,g] = hinfsyn(Pb,1,1,0,100,1e-2);

% == Transform the controller back to the original domain
[Akb,Bkb,Ckb,Dkb] = unpck(k) ;
[Ak,Bk,Ck,Dk] = bilin(Akb,Bkb,Ckb,Dkb,-1,'Sft_jw',[p1 p2]) ;
K = pck(Ak,Bk,Ck,Dk) ;
```

```
% == Check for the closed loop poles
Pcl = starp(P,K);
spoles(Pcl)

% == Check the hinfty norm, if it's " < 2 " --> robust stability
hinfnorm(sel(Pcl,(1:1),(1:1)))      %....>2 --> Robust stability failed
pause

% 2. Designing a Controller for Robust Stability and Performance
% == Check if Robust Performance is satisfied as it is
% ==== a. Settling time  due to the impulse disturbance
T = [0:0.01:20] ;
subplot(211) ;
[acl,bcl,ccl,dcl] = unpck(sel(Pcl,2,2));
sys = ss(acl,bcl,ccl,dcl) ;
[y,T] = impulse(sys,T);
plot(T,y); axis auto ; ylabel('x_2(t)') ;   grid ;


% ==== b. Control action  due to the impulse disturbance
% Need to modify the plant to have the displacement and control
% action available as performance outputs.
C1 = [C1;0 0 0 0];    C = [C1; C2] ;
D11 = [D11;0 0] ;  D12 = [D12;1] ; % Output u for performances spcs
D = [D11 D12; D21 D22] ;
Subplot(212) ;
P = pck(A, B, C, D);
Pcl = starp(P,K);
[acl,bcl,ccl,dcl] = unpck(sel(Pcl,3,2));
sys = ss(acl,bcl,ccl,dcl) ;
[u,T] = impulse(sys,T);             % The control action is large!!
plot(T,u); axis auto ; ylabel('u(t)') ; grid ;
pause
%
% Need to put a weight in the control action and displacement
%  To  guess  the  parameters  of  these  filters  look  at  the  frequency
response
close all ;
subplot(211)
omega = logspace(-2,4);
out_u = frsp(sel(Pcl,3,2), omega); % control freq response
vplot('liv,lm',out_u)
subplot(212)
out_x2 = frsp(sel(Pcl,2,2), omega); % displacement freq response
vplot('liv,lm',out_x2)
pause

% select cutoff freq for the control action
%   Note, using a second order weight to get sharper cut-off
Wu  =  nd2sys(conv([1  .8],[1  .8]),conv([1,100],[1,100]),7000);  %Wu  =
nd2sys([1 1],[1 100],5000);

% select cutoff freq for the displacement also at about 1 Hz
Wz = nd2sys([1 1],[1 1],1e-4);
%
```

```
% Plot the weights and the actual frequency responses
%
close all ;
outwx = frsp(Wz,omega); % output weight frequency response
outwu = frsp(Wu,omega); % control weight frsp
subplot(121)
vplot('liv,lm',outwx,out_x2,outwu,out_u)
legend('output weight','X_2(\omega)','control weight','U(\omega)')
pause
%
% Add a little bit of uncertainty weight to
% force the Hinfty norm to be less than 2
%
Wd = [0.5];
W = daug(Wd,Wz,Wu,1);
Pa = mmult(W,P);
Pa = minsys(Pa);
[Aa,Ba,Ca,Da] = unpck(Pa);

%       2.1 Hinfinty Controller Design
% == Transform the plant via a 'pole-shifting' bilinear transform:
[Ab,Bb,Cb,Db]=bilin(Aa,Ba,Ca,Da,1,'Sft_jw',[p1 p2]);

% == H-Inf Design:
Pb = pck(Ab,Bb,Cb,Db);
[k,g] = hinfsyn(Pb,1,1,0,1e10,1e-2);

% == Transform the controller back to the original domain
[Akb,Bkb,Ckb,Dkb] = unpck(k);
[Ak,Bk,Ck,Dk] = bilin(Akb,Bkb,Ckb,Dkb,-1,'Sft_jw',[p1 p2]);
K = pck(Ak,Bk,Ck,Dk);
minfo(K)
% == Check for the closed loop poles
Pcl = starp(P,K);
spoles(Pcl)

% == Check the hinfty norm, if it's < 2 --> robust stability
M = sel(Pcl,(1:1),(1:1)) ;
hinfnorm(M)

% now check the settling time and control action due to the impulse
disturbance
%figure(2)
subplot(222)
[acl,bcl,ccl,dcl]=unpck(sel(Pcl,2,2));  % Displacement
sys = ss(acl,bcl,ccl,dcl) ;
[y,T] = impulse(sys,T);
plot(T,y); axis auto ; ylabel('x_2(t)') ;   grid ;

subplot(224)
[acl,bcl,ccl,dcl]=unpck(sel(Pcl,3,2));  % Control Action
impulse(acl,bcl,ccl,dcl);
sys = ss(acl,bcl,ccl,dcl) ;
[u,T] = impulse(sys,T);
plot(T,u); axis auto ; ylabel('u(t)') ;     grid ;
```

### A.3 Code for the Mass Spring Damper Problem

```
%===========================================================%
% The Brennan Research Group, February 2006                 %
% Haftay Hailu, Ph.D Student                                %
% Sean Brennan (Asst. Prof. ), Advisor                      %

% Robust Stabilizing control synthesis for the              %
%                   Mass-Spring-Damper system               %
% User functions to be inlcluded, when running this code    %
%   1. minsys                                               %
%-----------------------------------------------------------*

% First create the generalized plant. Inputs w_k, w_c, w_m, u
%                                    Outputs z_k, z_c, z_m, y(=x)

clear all ;
close all ;
clc ;

% First consider the dimensioned system

A = [0 1;
    -1 -1] ;
B1 = [ 0 0 0;
    -1 -1 -1] ;                 % Inputs w_k, w_c, w_m
B2 = [0 1]' ;                   % Input Controller, u
C1 = [1 0; 0 1; -1 -1] ;        % Outputs z_k, z_c, z_m
C2 = [1 0] ;                    % output to the controller
D11 = [0 0 0; 0 0 0; -1 -1 -1] ;
D12 = [0 0 1]' ;  D21 = zeros(1,3) ;   D22 = [0]' ;
B = [B1 B2] ;   C = [C1; C2] ;  D = [D11 D12; D21 D22] ;
P = pck(A, B, C, D) ;

% Designing a Controller for Robust Stability
[gopt,K] = hinflmi(P,[1 1]) ;          % [nbr of measurements nbr of
controls]

% == Check for the closed loop poles
Pcl = starp(P, K) ;
spoles(Pcl)

% == Check the hinfty norm
M = sel(Pcl,(1:3),(1:3)) ; % System around the uncertainty loop
HinfNor = hinfnorm(M, 1e-6)
wc = HinfNor(3) ;
[Am, Bm, Cm, Dm] = unpck(M)
syms s M_tf Delta d_k d_c d_m
Delta = diag([d_k d_c d_m])
M_tf = Dm + Cm*inv(s*eye(3) - Am)*Bm
M_jwc = Dm + Cm*inv(j*wc*eye(3) - Am)*Bm
M_Delta = eye(3) - M_jwc*Delta
determ_M_Delta = det(M_Delta)
```

```
% The following results are determined from the equation
% "determ_M_Delta = 0". Since determ_M_Delta is a complex expression,
both
% the real and imaginary parts need to be equato to zero.

% for the above example for d_c = [-.5 .5]
dc = [-.5:0.05:.5] ;
dk = (1+4.49564*dc)/6377.234 ;
dm = 0.1 + 0.89921*dc ;
figure(1)
plot3(dc,dk,dm)

% The distance to the origin is
for i = 1:length(dc)
    d_origin(i) = sqrt(dc(i)^2 + dk(i)^2 + dm(i)^2) ;
end
figure(2)
plot(dc,d_origin)
grid
```

## A.4 Code for the Vehicle Autopilot Problem

```
%-----------------------------------------------------------*
% The Brennan Research Group, February 2006                 %
% Haftay Hailu, Ph.D Student                                %
% Sean Brennan (Asst. Prof. ), Advisor                      %

% Vehicle Autopilot: A Robust Simultaneous Control          %
%             through Parametric Adaptation                 %
%                                                           %
% User functions to be included, when running this code     %
%   1. GetNominalModel.m                                    %
%   2. Pi2p_1.m                                             %
%   1. minsys.m                                             %
%-----------------------------------------------------------*

clear all ;     close all ; clc ;

% Nominal Dimensionless Parameters and corresponding uncertainities

%ModelNo=1: Mean of Distribution, 2: around the scaled vehicle
ModelNo = 2 ;
[Pio,del_Pi] = GetNominalModel(ModelNo) ;   % Gets the nominal models
[po, del_p] = Pi2p_1(Pio, del_Pi)            % changes the pi's tp p's
(see thesis)

A = [0 1 0 0;
    0 -po(5) po(5) po(6);
```

```
    0 0 0 1;
    0 po(4) -po(4) -po(3)] ;
B1 = [0 0 0 0 0 0;
    0 1 0 0 -1 1;
    0 0 0 0 0 0;
    1 0 -1 1 0 0] ;         % Disturbances w_1...w_6
B2 = [0 po(2) 0 po(1)]' ;   % Controller, u = del_f
C1 = [0 0 0 0;
    0 0 0 0;
    0 0 0 1;
    0 1 -1 0;
    0 1 -1 0;
    0 0 0 1] ;              % Part of outputs z_1 ... z_6
D11 = zeros(6,6) ;
D12 = [1 1 0 0 0 0]' ;      % Part of outputs z_1 ... z_6

C2 = [1 0 0 0; 0 0 1 0] ;   % output to the controller, y_ and Si_
D21 = zeros(2,6) ;
D22 = [0 0]' ;

B = [B1 B2] ;
C = [C1; C2] ;
D = [D11 D12; D21 D22] ;
P = pck(A, B, C, D) ;

% 1. Designing a Controller for Robust Stability

% add the weighting due to the disturbance/uncertainty
Wz = diag(del_p) ;
W = daug(Wz, 1, 1); % Wz is the scaling/weights due to the
uncertainties
Pa = mmult(W,P);
Pa = minsys(Pa);
[Aa,Ba,Ca,Da] = unpck(Pa);

% == Transform the plant via a 'pole-shifting' bilinear transform:
q1 = -0.3 ;                                         % p1 controls settling
time;
q2 = -1000;                                     % p2 controls bandwidth
[Ab,Bb,Cb,Db] = bilin(Aa, Ba, Ca, Da,1,'Sft_jw',[q1 q2]);

% == H-Inf Design:
Pb = pck(Ab,Bb,Cb,Db);
[k,g] = hinfsyn(Pb,2,1,0,100,1e-2);

% == Transform the controller back to the original domain
[Akb,Bkb,Ckb,Dkb] = unpck(k) ;
[Ak,Bk,Ck,Dk] = bilin(Akb,Bkb,Ckb,Dkb,-1,'Sft_jw',[q1 q2]) ;
K = pck(Ak,Bk,Ck,Dk) ;

% == Check for the closed loop poles


% == Check the hinfty norm, if it's " < max " --> robust stability
Pcl = starp(P,K) ;
```

```
hinfnorm(sel(Pcl,(1:6),(1:6)))

% 2. Designing a Controller for Robust Stability and Nominal
Performance

% == Check if Robust Performance is satisfied in the above design
% First modify the model to include external disturbances F_, M_
% and to output y, phi and u for performance specs.
B1 = [B1 [0 1 0 0; 0 0 0 1]'] ;
C1 = [C1;
     1 0 0 0;
     0 0 1 0;
     0 0 0 0] ; % outputing x1, x3 and u for performance weights
D11 = zeros(9,8) ;
D12 = [D12; 0; 0; 1] ;
D21 = zeros(2,8) ;
B = [B1 B2] ;
C = [C1; C2] ;
D = [D11 D12; D21 D22] ;
P = pck(A, B, C, D) ;
Pcl = starp(P,K) ;

T = [0:0.01:20] ;
subplot(221) ;
[acl,bcl,ccl,dcl] = unpck(sel(Pcl,7,7));    % check y due to impulse F
sys = ss(acl,bcl,ccl,dcl) ;
[y,T] = impulse(sys,T);
plot(T,y); axis auto ; ylabel('y(t)') ;   grid ;

subplot(222) ;
[acl,bcl,ccl,dcl] = unpck(sel(Pcl,8,8));
sys = ss(acl,bcl,ccl,dcl) ;
[y,T] = impulse(sys,T);
plot(T,y); axis auto ; ylabel('\psi(t)') ;   grid ;  % check psi due to
impulse M

subplot(223) ;
[acl,bcl,ccl,dcl] = unpck(sel(Pcl,9,7));
sys = ss(acl,bcl,ccl,dcl) ;
[y,T] = impulse(sys,T);
plot(T,y); axis auto ; ylabel('u_F(t)') ;   grid ;   % check controller
due to impulse F

subplot(224) ;
[acl,bcl,ccl,dcl] = unpck(sel(Pcl,9,8));
sys = ss(acl,bcl,ccl,dcl) ;
[y,T] = impulse(sys,T);
plot(T,y); axis auto ; ylabel('u_M(t)') ;   grid ;   % check controller
due to impulse M


% Need to put a weight in the control action and displacement
% To guess the parameters of these filters look at the frequency
response
figure(2)
```

```
omega = logspace(-2,4);
subplot(221)
out_y1  =  frsp(sel(Pcl,7,7),  omega);  %  latera  displacement  (y)  freq
response
vplot('liv,lm',out_y1) ; ylabel('Y') ; grid ;
subplot(222)
out_y3 = frsp(sel(Pcl,8,8), omega); % yaw angular displacement () freq
response
vplot('liv,lm',out_y3) ; ylabel('\psi') ; grid ;
subplot(223)
out_uF = frsp(sel(Pcl,9,7), omega); % control freq response
vplot('liv,lm',out_uF) ; ylabel('F') ; grid ;
subplot(224)
out_uM = frsp(sel(Pcl,9,8), omega); % control freq response
vplot('liv,lm',out_uM) ; ylabel('M') ; grid ;

% select cutoff freq for the displacement y at 0.01 rad/sec
Wz7 = 1 ;%nd2sys([1 1000],[1 100],10);
Wz8 = 1 ;%nd2sys([1 1000],[1 100],1);;

% select cutoff freq for the control action
% Wu = nd2sys([1 0.8],[1 100],100); %---> works with some overshoots
% Wu = nd2sys([1 0.3],conv([1 0.02], [1 300]),4000); %--> doesn't work
Wu = nd2sys(conv([1 0.1],[1 2]),conv([1 50],[1 100]),50000); %---> the
        %  best  so  far  along  with,  Wz7  =  Wz8  =  nd2sys([1  1000],[1
1],0.01); ...
        % it also works good for scaled vehicle model with aturation
        %  the  Hinfnorm  is  about  12.08  which  allows  a  perturbation  of
0.0833

% Plot the weights and the actual frequency responses
%
figure(2)
outwz7 = frsp(Wz7,omega); % output weight frequency response
outwz8 = frsp(Wz8,omega); % output weight frequency response
outwz9 = frsp(Wu,omega); % control weight frsp
subplot(221)
vplot('liv,lm',outwz7,out_y1)
legend('Output weight','Y_1(\omega)')
subplot(222)
vplot('liv,lm',outwz8,out_y3)
legend('Output weight','Y_3(\omega)')
subplot(223)
vplot('liv,lm',outwz9,out_uF)
legend('control weight','U_F(\omega)')
subplot(224)
vplot('liv,lm',outwz9,out_uM)
legend('control weight','U_M(\omega)')

% Put together all the weights and design a controller

Wz = diag(del_p);
W = daug(Wz, Wz7, Wz8, Wu, 1, 1); % Wz, Wu are the scaling/weighting
Pa = mmult(W,P);
Pa = minsys(Pa);
```

```
[Aa,Ba,Ca,Da] = unpck(Pa);

%       2.1 Hinfinty Controller Design

% == Transform the plant via a 'pole-shifting' bilinear transform:
[Ab,Bb,Cb,Db]=bilin(Aa,Ba,Ca,Da,1,'Sft_jw',[q1 q2]);

% == H-Inf Design:
Pb = pck(Ab,Bb,Cb,Db);
[k,g] = hinfsyn(Pb,2,1,0,1000000000,1e-2);

% == Transform the controller back to the original domain
[Akb,Bkb,Ckb,Dkb] = unpck(k);
[Ak,Bk,Ck,Dk] = bilin(Akb,Bkb,Ckb,Dkb,-1,'Sft_jw',[q1 q2]);
K = pck(Ak,Bk,Ck,Dk);
minfo(K)
% == Check for the closed loop poles
Pcl = starp(P,K);
spoles(Pcl)

% == Check the hinfty norm, if it's " < max " --> robust stability
hinfnorm(sel(Pcl,(1:6),(1:6)))


% now check the settling time and control action due to the impulse
disturbances
figure(4)   % response to a lateral impulse force input
subplot(211) ;
[acl,bcl,ccl,dcl] = unpck(sel(Pcl,7,7));
sys = ss(acl,bcl,ccl,dcl) ;
[y,T] = impulse(sys,T);
plot(T,y); axis auto ; ylabel('y(t)') ;   grid ;

subplot(212) ;
[acl,bcl,ccl,dcl] = unpck(sel(Pcl,9,7));
sys = ss(acl,bcl,ccl,dcl) ;
[y,T] = impulse(sys,T);
plot(T,y); axis auto ; ylabel('u_F(t)') ;   grid ;

figure(5)   % response to a rotational impulse moment input
subplot(211) ;
[acl,bcl,ccl,dcl] = unpck(sel(Pcl,8,8));
sys = ss(acl,bcl,ccl,dcl) ;
[y,T] = impulse(sys,T);
plot(T,y); axis auto ; ylabel('\psi(t)') ;   grid ;

subplot(212) ;
[acl,bcl,ccl,dcl] = unpck(sel(Pcl,9,8));
sys = ss(acl,bcl,ccl,dcl) ;
[y,T] = impulse(sys,T);
plot(T,y); axis auto ; ylabel('u_M(t)') ;   grid ;

% Load results into Simulink
[Ak1,Bk1,Ck1,Dk1] = unpck(K);
```

```
[Ak2,Bk2,Ck2,Dk2] = unpck(minsys((K)));

% 3. Transform the controller to dimensional domain and Simulate

% Nominal parameters, use these values for ModelNo = 2 above
L = 0.655 ;    % m
U = 2.5 ;      % m/s
M = 11.4 ;

a = 2*Pio(1)*L ;          % [m], Pio(1) = Pi1_nominal
b = L-a ;               % [m]
Caf = Pio(2)*M*U^2/L ;     % [N/rad], Pio(2) = Pi3_nominal  ;
Car = Pio(3)*M*U^2/L ;  ;   % [N/rad], Pio(3) = Pi4_nominal  ;
Iz = Pio(4)*M*L^2 ;     % [kgm^2], Pio(4) = Pi5_nominal

% The dimensional transformation matrices of the plant
Mx = diag([1/L 1/U 1 L/U]) ;
My = diag([1/L 1]) ;
Mu = diag([1]) ;
Nt = U/L ;          % time scale

% The dimensional transformation matrices of the controller
Mkx = Mx ;
Mky = Mu ;
Mku = My ;

% The dimensionless plant model from above
Ad = A ;
Bd = B2 ;
Cd = C2 ;
Dd = D22 ;

% Transformation of the plants from dimensionless to dimensional domain
Am1 =  Nt*inv(Mx)*Ad*Mx ;
Bm1 =  Nt*inv(Mx)*Bd*Mu ;
Cm1 =  inv(My)*Cd*Mx ;
Dm1 =  inv(My)*Dd*Mu ;

% Unpack the dimensionless controller designed above
[Akd, Bkd, Ckd, Dkd] = unpck(minsys(K)) ;
%  Modify  the  dimensional  transformation  matrixes  to  consider  the
dynamics
% of the performance weights
[RowAkd, ColAkd] = size(Akd) ;
[RowMkx, ColMkx] = size(Mkx) ;
Mkx = [Mkx zeros(RowMkx, ColAkd-ColMkx);
    zeros(RowAkd-RowMkx, ColMkx) eye(RowAkd-RowMkx,ColAkd-ColMkx)];

%  Transformation  of  the  the  dimensionless  controller  to  dimensional
domain
Akm =  Nt*inv(Mkx)*Akd*Mkx ;
Bkm =  Nt*inv(Mkx)*Bkd*Mku ;
Ckm =  inv(Mky)*Ckd*Mkx ;
Dkm =  inv(Mky)*Dkd*Mku ;
```

```
% Compare the transformed plant (This is only for verification)
f1 = Caf+Car ;
f2 = a*Caf - b*Car ;
f3 = a^2*Caf+b^2*Car ;
Am2 = [0 1 0 0; 0 -f1/(M*U) f1/M -f2/(M*U); 0 0 0 1; 0 -f2/(Iz*U) f2/Iz
-f3/(Iz*U)]
Bm2 = [0 Caf/M 0 a*Caf/Iz]' ;
Cm2 = [1 0 0 0; 0 0 1 0] ;
Dm2 = [0 0]' ;
VehicleAutopilotSMLK20060224;
```

# VITA: Haftay Hailu

**EDUCATION:**

| | |
|---|---|
| 2002-2006 | THE PENNSYLVANIA STATE UNIVERSITY, University Park, PA<br>**Ph.D. in Mechanical Engineering**, *May 2006*<br>**M.S. in Electrical Engineering**, *May 2006* |
| 1999-2001 | LOUISIANA STATE UNIVERSITY, Baton Rouge, LA<br>**M.S. in Mechanical Engineering**, *December 2001* |
| 1992-1997 | ADDIS ABABA UNIVERSITY, ADDIS ABABA, ETHIOPIA<br>**B.Sc. in Mechanical Engineering**, *July 1997* |

**PROFESSIONAL EXPERIENCE:**

| | |
|---|---|
| Jul. 2005 – May 2006 | National Science Foundation GREATT Fellow<br>The Pennsylvania Transportation Institute, The Pennsylvania State University.<br><br>Penn State Challenge X group leader<br>The Pennsylvania Transportation Institute |
| May 2005 – Jun. 2005 | Graduate Teaching Fellow for ME 23: Introduction to Thermal and Fluid Sciences<br>Department of Mechanical & Nuclear Engineering, The Pennsylvania State University. |
| Aug. 2002 – May 2005 | Teaching Assistant for ME 50: Machine Dynamics, ME 54: Dynamics of Mechanical Systems and ME 82: Mechanical Engineering Measurements<br>Department of Mechanical Engineering, The Pennsylvania State University |
| Aug. 1999 – Dec. 2001 | Teaching and Research Assistant<br>Department of Mechanical Engineering, Louisiana State University |
| Jul. 1997 – Aug. 1999 | Assistant Lecturer (Jul. 1997- Aug. 1998) Graduate Assistant (Sep. 1998- Aug. 1999)<br>Department of Mechanical Engineering, Faculty of Technology, Addis Ababa University<br><br>Engineering Consultant for Design and CADD Applications (Part-time)<br>Dan Technocraft P.L.Co., Addis Ababa, Ethiopia<br>Salvalore De Vita PLC, Addis Ababa, Ethiopia<br>Selam Technical and Vocational Center |

**AWARD AND HONORS:**

***NSF-GREATT Fellowship:*** The Pennsylvania Transportation Institute, The Pennsylvania State University, July 2005, University Park, PA.

***Graduate Teaching Fellowship:*** Department of Mechanical and Nuclear Engineering, The Pennsylvania State University, May 2005, University Park, PA.

***Outstanding Teaching Assistant Award:*** Department of Mechanical Engineering, Louisiana State University, December 2001, Baton Rouge, LA.

***Graduate School Tuition Award:*** Louisiana State University, Aug. 2000, Baton Rouge, LA 70803