ABSTRACT

ANDJELKOVIC, IVAN V. Auxiliary Signal Design for Fault Detection for Nonlinear Systems: Direct Approach. (Under the direction of Professor Stephen L. Campbell).

The main task of active fault detection is to design an auxiliary signal which acting on the system will reveal to the observer a potential fault of the system. There are numerous results that implement techniques of optimal control to calculate an auxiliary signal. However, the techniques and methods are almost exclusively designed for linear systems, while nonlinear systems are treated through linearization. In this thesis, we are providing a novel way of directly approaching nonlinear systems.

We will start with a brief overview of the areas of the fault detection, optimal control, basic terminology and principles of active fault detection and previous research. Then we will present the novel p-norm approach enabling us to solve nonlinear problems directly using optimization. We will develop a direct optimization formulation for active fault detection for linear and nonlinear systems with additive uncertainty. We will present some test examples and point out the advantages and disadvantages of our new p-norm approach. Several illustrative examples will be presented and analyzed for a deeper understanding of underlying problems involved with nonlinear systems. We are hoping that this thesis will give guidelines for future users and researchers of how to approach active fault detection on nonlinear systems.

Linear systems with model uncertainty have already been analyzed using the Riccati equations. Here we will develop a direct optimization formulation. After some test examples, we will solve several types of problems that cannot be solved using the Riccati approach, namely problems that contain additional constraints (soft or hard) on states of the system or on auxiliary signals. The quality of an auxiliary signal is usually measured by some cost function. We will examine the influence of several cost functions on our auxiliary signal design.

Auxiliary Signal Design for Fault Detection for Nonlinear Systems: Direct Approach

by
Ivan V. Andjelkovic

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fullfillment of the
requirements for the Degree of
Doctor of Philosophy

Applied Mathematics

Raleigh, North Carolina

2008

APPROVED BY:

Dr. Pierre A. Gremaud	Dr. Negash Medhin
Dr. Stephen L. Campbell Chair of Advisory Committee	Dr. Ralph Smith

DEDICATION

To Lisa.

BIOGRAPHY

Ivan Andjelkovic was born in Belgrade, Serbia. He graduated from the University of Belgrade in 1996 with a BS degree in Electrical and Computer Engineering. He obtained his MS degree in Electrical and Computer Engineering from Drexel University in Philadelphia, Pennsylvania in 1999. His MS thesis was "Trellis-Coded Modulation with Applications to Unequal Error Protection", written under the patient and expert guidance of his adviser Dr. Maja Bystrom. He then worked for 2 years at Unisys in Malvern, Pennsylvania as a hardware engineer. After the economic downturn in 2001, he worked as an astronomy instructor and a sales person. In 2003, he returned to his studies at North Carolina State University as a PhD candidate in Applied Mathematics under the supervision of Dr. Stephen L. Campbell. He also has two beautiful daughters, Lenore and Adrijana.

ACKNOWLEDGMENTS

First, I would like to thank my adviser Stephen L. Campbell. Without his experience, words of wisdom and knowledge my research would not have been possible.

I would like to thank my committee members, Dr. Campbell, Dr. Smith, Dr. Gremaud and Dr. Medhin for their valuable inputs and time.

Life would be quite difficult without funding, hence I would like to thank the Department of Mathematics at NCSU for their financial support through a Teaching Assistantship. I would also thank NSF and NAVSEA for funding for my Research Assistantship.

Like in most research, there are people who enriched my doctoral experience, either through the exciting project at NAVSEA directed by Dr. Kimberly Clark, by introducing me to the software tool and friendship like SOCS guru Dr. Anna Engelsone or by providing me with the additional results like Dr. Kelly Sweetingham and Dr. Dongkyoung Choe. Thank you!

The numerical analysis seminar group helped me develop my latex and presentational skills. Thank you all.

I want to thank my longest supporters, my mom, and my brother. Though they are across the ocean, they are always next to me. I also want to mention my dad, who is always in my thoughts.

Finally, to give meaning to everything, there are my three muses, my beautiful, smart, perfect mother, wife and partner Lisa, and our two little angels, Lena and Jana.

TABLE OF CONTENTS

LIST OF TABLES . . . . **vii**

LIST OF FIGURES . . . . **viii**

1 Introduction **1**
1.1 Passive vs Active Fault Detection Approach 2
1.2 Model Preparation for Active Fault Detection 5
1.3 Objective of the Thesis 12

2 Active Fault Detection-Previous Results **13**
2.1 Optimal Control Background 13
2.2 Necessary Conditions 18
2.3 Modeling and Basic Principles of Active Fault Detection 20
2.3.1 Active Fault Detection on Linear Systems with Additive Uncertainty 22
2.3.2 Formulation of the Linear Systems with Model and Additive Uncertainty 29
2.3.3 Nonlinear Results and Approach 31
2.3.4 Optimal Control Solver SOCS and Other Software 33

3 Active Fault Detection for Nonlinear Systems **36**
3.1 Definition of the Problem 36
3.1.1 p-norm Approach 39
3.2 Definition of Problems with Additive Noise Only 43
3.2.1 Nonlinear Problems 43
3.2.2 Linear Problems 50
3.3 Linear Problems with Model and Additive Uncertainty 53
3.3.1 Software Implementation 57
3.3.2 Test Examples 59
3.4 Noise Bound Setup 66

4 Examples and Analysis **69**
4.1 Example 1 69
4.1.1 Linearized Problem with L^2 Norm as a Cost Function 73
4.1.2 Nonlinear Problem with L^2 Norm as a Cost Function 75
4.1.3 Nonlinear Problem with General Cost Function 80
4.2 Example where Linearization does not Work 90

4.3 Real Physical System Example 90
4.4 Model Uncertainty Example 95
4.5 Other Extensions of Linear Model Uncertainty Examples 98
4.5.1 Hard Bound on the Auxiliary Signal v 100
4.5.2 Soft Bound on the Auxiliary Signal v 101
4.5.3 Bound on the States 105

5 Summary of Contributions 108

6 Future Work 112
6.1 p-norm Approach for Model Uncertainty 112
6.2 Delay Implementation in SOCS 114
6.3 Other Extensions 115

Bibliography 118

LIST OF TABLES

Table 3.1 Cost and noise bound of auxiliary signal for linear Example 4.2.2 of [7] 42

Table 3.2 SOCS time required to calculate auxiliary signal for linear Example 4.2.2 of [7] 42

Table 3.3 Parameter comparison of SOCS and Scilab runs for $U = W = 0$ 60

Table 3.4 Parameter comparison of SOCS and Scilab runs for $U = W = I$ 62

Table 3.5 Parameter comparison of different solutions for v 66

Table 4.1 Calculated cost for $n_i = 0.1$, $P_i = 10$ 75

Table 4.2 Cost of L^2 norm cost function 77

Table 4.3 SOCS time required to find v 78

Table 4.4 Noise bounds when using v_{lin} and v_p 79

Table 4.5 Cost of linearized model for various values of U 83

Table 4.6 Cost of nonlinear model for various values of U 83

Table 4.7 Algorithm performances for $T = 10$ and various values of U 88

Table 4.8 Performance comparison 93

Table 4.9 Noise bound for various initial guesses 94

Table 4.10 Parameters for various g and $P = I$ 96

Table 4.11 Parameters for various g and $P = 0.0001I$ 97

Table 4.12 Parameters for various g and $P = 0$ 99

LIST OF FIGURES

Figure 1.1 Compressor - tank system 6

Figure 2.1 Input-output set separation by proper signal v 24

Figure 3.1 Comparison of p-norm and max norm 40

Figure 3.2 Aux signals for linear Example 4.2.2 of [7] 43

Figure 3.3 Comparison of signals v obtained by SOCS and Scilab for $U = W = 0$ 61

Figure 3.4 Comparison of signals v obtained by SOCS and Scilab for $U = W = I$ 62

Figure 3.5 The best signals v found 64

Figure 3.6 The suboptimal signal 1 64

Figure 3.7 Suboptimal signal 2 65

Figure 3.8 Comparison of the signals v 65

Figure 4.1 V optimal for various linearized cases using β approach, P=10 74

Figure 4.2 Comparison of v for T=1, n=0.1, P=10 76

Figure 4.3 Comparison of v for T=5, n=0.1, P=10 77

Figure 4.4 Effect of U on v for linearized model 84

Figure 4.5 Effect of U on v for nonlinear model 84

Figure 4.6 Comparison between v_p and v_{lin} for $U = 0$ and $U = 10$ 85

Figure 4.7 Equilibrium analysis of linearized and nonlinear nonfaulty model 86

Figure 4.8 Comparison of signal ψ for linearized and nonlinearized models when $U = 1$ 87

Figure 4.9 Test signal v on $[0, 10]$ for different U values 88

Figure 4.10 Proper signal v for Example 4.2 found using 2-norm 91

Figure 4.11 Comparison of v_{lin} and v_p for Example 4.3 93

Figure 4.12 Auxiliary signal v for various values of g, and $P = I$ 96

Figure 4.13 Auxiliary signal v for various values of g, and $P = 0.0001I$ 97

Figure 4.14 Auxiliary signal v for various values of g, and $P = 0$ 99

Figure 4.15 v with hard bound on auxiliary signal v, $g = 0$ 101

Figure 4.16 v with hard bound on auxiliary signal v, $g = 0.05$ 102

Figure 4.17 v with soft and hard bounds on auxiliary signal v, $g = 0$ 103

Figure 4.18 v with soft and hard bounds on auxiliary signal v, $g = 0.05$ 104

Figure 4.19 Noncontinuous v if additional hard bounds are not used 104

Figure 4.20 v comparison for bounds on states x_i 106

Figure 4.21 x_i comparison for bounds on states x_i 106

Figure 6.1 SOCS implementation of problems with delay 116

Chapter 1

Introduction

First we would like to give a brief overview of the thesis.

In chapter 1, basic definitions and comparison of active vs passive fault detection will be given. We will explain motivation for active fault detection. An example representing issues that might occur during applications of active fault detection will be presented, with emphasis on the modeling issues. The objectives of the thesis will be defined as well, such as the focus on nonlinear problems and developing an algorithm that attacks nonlinear problem directly.

In chapter 2, basic optimal control background will be given together with a literature overview. Model based fault detection will be presented with focus on modeling of additive and model uncertainty. Active fault detection theory and results so far will be described together with the basic theory of linearized approach. An overview of the current solutions for nonlinear problems will be given, and the software package used for examples and simulations described.

Chapter 3 will present the novel approach to using active fault detection on systems with additive noise and model uncertainty. The idea of using a p-norm will be introduced. Derivation of necessary conditions for nonlinear systems with additive noise and linear systems with additive noise using p-norm will be presented. We will briefly analyze the impact of using a p-norm, leaving detailed analysis for chapter

4. Finally, we will focus on linear problems with model uncertainty, derive necessary conditions and develop an algorithm that can accommodate various cost functions.

In Chapter 4, several examples with additive and model uncertainty will be presented with detailed analysis. We will focus on strengths and weaknesses of our approach in comparison with the standard β approach used for linearized systems. We will also analyze how the specific nonlinearity affects our solution. The analysis will be done for different cost functions (namely L^2 norm and amount of disturbance of the system). Some examples with additional constraints will be presented as well.

Chapter 5 will give an overview of contributions during my PhD research and chapter 6 will give guidance for future work.

1.1 Passive vs Active Fault Detection Approach

Broadly speaking, everything that surrounds us can be treated as a system, from the ones that can be represented as simple as a free falling ball, to the vastly complex ones like Earth ecology or the design of space shuttles.

Unfortunately, it is a fact of nature, that no matter what system we observe, given enough time, something will go wrong with it. If it is a "permanent interruption of a system's ability to perform a required function under specified operating condition" [27], we are referring to it as a failure (impact of hammer on a wrist watch). If it is "an unpermitted deviation of at least one characteristic property (feature) of the system from the acceptable, usual, standard condition", then we are referring to it as a fault (car with leaking exhaust pipe).

Naturally, we would want to do all that is in our power to keep our systems working properly and/or to minimize effects of faults and failures. Beside just not having a fully operating system (which is in itself significant), occurrence of a fault or failure can have catastrophic consequences (Chernobyl meltdown), can increase cost of repair (changing oil in a car is less expensive than replacing the engine) and lead to the other faults and failures (bad fuse leaves appliances vulnerable to the power surges). That desire to keep systems operating properly led to research, development and application of methods to supervise, detect, diagnose and manage faults and failures. In this thesis, the focus will be on the fault detection and, to a certain extent, diagnosis.

The traditional fault detection approach is using a passive fault detection. Passive fault detection is well established and well represented in everyday life, like all kinds of warning lights, fire and smoke alarms etc. The idea is to observe (measure or estimate) certain parameters or states of the system, and if those parameters or states deviate enough from the expected values (break thresholds), the fault would be declared. Thresholds themselves are typically chosen using a probabilistic approach to the system, i.e. we find that if a parameter is out of specific bounds, there is a small probability that the observed part of the system is operating to our expectations. There are numerous references on research and implementation of this approach [27], [5], [28], [39]. While extremely useful, there are some potential issues with this approach.

1. Complexity of the system can make observation of all parameters impossible.

2. Observation of parameters in itself increases the complexity of the system, and increased complexity increases the possibility of a fault. We need equipment like sensors and alarm mechanisms to perform observations and a fault in that equipment can destroy our passive fault detection scheme. If the lamp on the dashboard of our car that indicates overheating of the engine is burned out, we will not have valuable information of the state of the engine of our car leading to potential accidents.

3. One of the desired properties and a common requirement of today's complex and expensive systems is robustness. We want our system to operate even if something goes wrong. Robustness can be created by parallel circuitry or mod-

ules, systems of observers and controllers, specially designed feedback loops, etc. While having excellent methods to provide robustness is a great achievement of engineering and science, it also had an undesired effect of straining one part of the system if something is wrong with another part of the system, while at the same time, our observed parameters might remain within the valid range. That additional strain can lead to the more massive and expensive collapses of the system in the future, present potential danger, and reduce operational time of the equipment.

What is common for all passive methods is that they are not acting on the system. In this thesis we are focusing on alternative, active fault detection.

The basic idea of active fault detection is that instead of just observing the system we are going to act on it by creating an additional, "auxiliary" input signal. The purpose of this signal is to disturb the system just enough so we can conclude, based on the output, if any part of the system is at fault. Such signals are called proper. The "best" proper signal is called the "optimal proper" auxiliary signal. We have to keep in mind that during the testing, we want to avoid causing a big disruption in the systems operation or being the cause of the failure. Hence "the best" criterion is usually focused on keeping the effect of the test signal small. The actual criteria depends on the system itself and the designer of the testing procedure. Common choices are:

1. The smallest proper signal in some sense, usually the L^2 norm.

2. The signal that affects the system the least.

It is a good engineering practice to use the idea of active fault detection during the testing procedure of system design, as well as during the scheduled checkups and maintenance of the system. While our methods can be used on these occasions as well, our focus is on the applications of active fault detection during the regular operations of the system with minimal interruptions.

For our approach to active fault detection, we need to know a mathematical model of the system. The quality of the model is crucial in determining how good of a solution we can get. We can find the model either experimentally or by deriving it from physical laws that govern our system. In the following example we will present some modeling issues that can be encountered.

1.2 Model Preparation for Active Fault Detection

In this section, we will present an example of a fault detection setup on a real life system to give the reader a brief overview of the active fault detection approach and the issues that can arise during the implementation, with a strong emphasis on modeling. It represents the modified report that Dr. Campbell and I wrote regarding an ongoing project with NAVSEA [19].

The system consists of a modified TC-20T compressor with tank and 3eti sensors measuring pressures, temperatures and vibrations on various points of the system in figure 1.1. Observed parameters, components and properties of the system are:

1. Valve 1 between compressor and tank, Vlv1
2. Compressor auto-starts at 55 psi and stops at 75 psi
3. 3 separate vibration sensors placed on the head of the compressors motor
4. One vibration sensor placed at the foot of the compressor
5. Pressure measured at the output of the compressor P_h, input to the tank P_t, and output of the tank P_{out} which will also be referred to as the regulator pressure
6. Temperature of the compressor, head of the compressor, and tank (metal itself, not temperature of the air in it) are also measured
7. 3 potential leak places, L1, L2 and L3 (focus on L1)

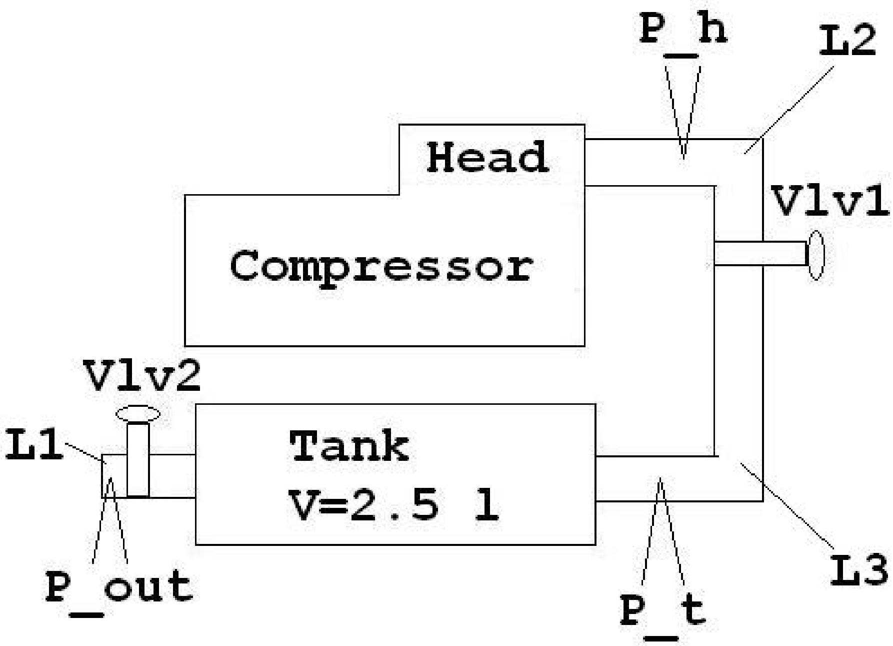

Figure 1.1: Compressor - tank system

8. Vlv2 which is a computer controlled valve 2

The basic question is to find out whether the tank leaks or not. The passive approach would observe the pressure in the tank and signal leakage if the pressure drops bellow a certain value. For small leaks, that can take a significant amount of time. The problem with this approach is that during that long period of time, temperature changes are affecting the pressure in the tank. When the compressor is on, the temperature increases. When the compressor is off, the temperature decreases, which in turn decreases pressure. This effect can mask the decrease in pressure due to the leakage. Also in field use there would be varying loads put on and long term removal from operation is not desirable. The temperature effect can be caused by the environment as well.

The active fault approach is to use control over compressor or valves Vlv1 or Vlv2, and together with observation of outputs P_h,P_t and P_{out} conclude whether tank is leaking or not. To be able to do that, finding the appropriate mathematical model of the system is of utmost importance. Inspired by [37], we modeled the above system focusing on the amount of air entering and leaving the tank measured in moles. Let M_G be the moles of air in the tank (pipes neglected), then:

$$\frac{dM_G}{dt} = F_G - G \tag{1.1}$$

where F_G is molar feed flow rate of air through valve 1 and G is molar outflow of air through valve 2. In the tank we assumed that ideal gas law is valid and that the molar flow is proportional to the pressure difference.

$$V = M_G \frac{RT}{P} \tag{1.2a}$$

$$G = k_{G2}\kappa_2(P_t - P_{out}) \tag{1.2b}$$

where T is the temperature of the air, V volume of the tank, R is constant, k_{Gi}'s are coefficients of valve i and κ_i's represent openings of valve i. L1, L2 and L3 represent potential leakage points with corresponding molar flow rate of F_{L1}, F_{L2} and F_{L3}. Note that since there is no flow if there is no pressure difference, if the relationship is smooth, then one can argue for (1.2b) for "small" pressure differences via a Taylor approximation. In the following formulas, we are going to use both, Leibnitz notation $\frac{d}{dt}$ and standard notation $\dot{()}$ for first derivatives.

We can replace M_G and G in (1.1) using equations (1.2a) and (1.2b) to get:

$$\frac{d\frac{P_tV}{RT}}{dt} = F_G - k_{G2}\kappa_2(P_t - P_{out}) \tag{1.3}$$

Using the quotient rule and the fact that V and R are constant, we get after a little simplifying:

$$\dot{P}_t = \frac{\dot{T}}{T}P_t - \frac{RTk_{G2}\kappa_2}{V}P_t + \frac{RTk_{G2}\kappa_2}{V}P_{out} + \frac{RT}{V}F_G \tag{1.4}$$

We have temperature sensors but they measure the temperature of the tank wall, compressor head, etc. We do not have any direct temperature measurements of air. Also, the data so far suggest that tank temperature does not vary greatly since the temperature of the tank wall only rises about 10%. In a larger tank one would expect the temperature effect to be even less. However, the temperature may be important, so we will monitor this. Note that the $\dot{T}$ term in (1.4) occurs in front of P_t. Thus we could view $\dot{T}$ as being included if we allow for model uncertainty in the coefficient of P_t.

Since the molar flow on the output of valve 2, G, is modeled as being directly proportional to the pressure difference with the proportionality constant k_{G2}, we will assume that we can apply proportionality on all other flows. Using this logic, we can model F_G as a flow through valve 1, that is,

$$F_G = k_{G1}\kappa_1(P_h - P_t) \tag{1.5}$$

This model neglects compressibility of the gas and ignores turbulation effects in the case of small openings. Keeping the above limitations in mind, we can use the same proportional model for leakage simulation.

The discussion so far only concerned the basic inflow and outflow. Let us rewrite the above equations now to include only leakage L1 as a fault for our system ($G = F_{vlv2}, F_G = F_{vlv1}$) under assumption that T is constant. This assumption is not in contradiction with the premises from the set up of the problem, since testing time will be relatively short.

$$\frac{dM_G}{dt} = F_{vlv1} - F_{L1} - F_{vlv2} \tag{1.6}$$

Modeling flows as being proportional to pressure difference we get:

$$\frac{dM_G}{dt} = k_{V1}\kappa_1(P_h - P_t) - k_{L1}(P_{out} - P_{atm}) - k_{V2}\kappa_2(P_t - P_{out}) \tag{1.7}$$

Including equation (2) as earlier, we get

$$\dot{P}_t = \frac{RT}{V}(k_{V1}\kappa_1(P_h - P_t) - k_{L1}(P_{out} - P_{atm}) - k_{V2}\kappa_2(P_t - P_{out})) \tag{1.8}$$

This equation only deals with the flows into and out of the tank.

We will also consider P_h as a state. Flow out of the part of the pipe on that pressure is through valve 1, hence flow out will be the same as F_G. Inflow comes from the compressor which is considered to be on or off.

We have no direct pressure measurements inside the compressor. The compressor in our experimental setup is a piston type compressor. Its controller has it either on or off. The compressor performs work in increasing the air pressure from P_{atm} to P_h. A typical model for the work of gas compression in one cycle is that this work is proportional to $(\frac{P_h}{P_{atm}})^a - 1$ where $0 < a < 1$. The rate of work depends on compressor rpm, that is how often the piston pushes the air per minute. As P_h increases, the amount of work required for each compression stroke increases. If we assume that rate of work (power) is constant and that the compressor delivers the same number of moles per cycle which is reasonable since it intakes the same amount of outside air on each stroke, we have

$$rpm * \left(\left(\frac{P_h}{P_{atm}} \right)^a - 1 \right) = constant \tag{1.9}$$

so that

$$rpm = \frac{constant}{\left(\frac{P_h}{P_{atm}} \right)^a - 1} \tag{1.10}$$

Assuming that each push of the piston sends the same amount of air out of the compressor, that is the cylinder is fully emptied, the rate of pressure change is proportional to rate of mole flow, which is in turn proportional to rpm. Since P_{atm} is constant we have:

$$rate\ of\ pressure\ change = const * rpm = \frac{constant}{(\frac{P_h}{P_{atm}})^a - 1} = \frac{1}{c_1 P_h - c_2} \tag{1.11}$$

This equation will model flow from the compressor. Put together our differential equation for state P_h is

$$\dot{P}_h = \sigma \frac{1}{c_1 P_h - c_2} - k_{G1} \kappa_1 (P_h - P_t) \tag{1.12}$$

σ has either value 0 if compressor is off, and value 1 if compressor is on.

Simplifying expressions by taking $\frac{RT}{V}$ as a constant and combining it with other constants it multiplies we get the nonlinear model with state equations

$$\dot{P}_t = k_{V1}\kappa_1(P_h - P_t) - k_{L1}(P_{out} - P_{atm}) - k_{V2}\kappa_2(P_t - P_{out}) + N_1\nu \quad (1.13a)$$

$$\dot{P}_h = \sigma\frac{1}{c_1P_h - c_2} - k_{G1}\kappa_1(P_h - P_t) + N_2\nu \quad (1.13b)$$

Output equation:

$$y = P + M\nu \quad (1.13c)$$

and bound on uncertainty:

$$\delta(P(0) - P_{t0})^2 + \int_0^T \nu^2\, dt < d \quad (1.13d)$$

where $P = [P_t\ P_h]^T$ and δ is a weighting constant. Notice that we have control over openings of the valves, hence k_{V1} and k_{V2} are our auxiliary controls. The task of the active fault detection will be to determine the best k_{V1} and/or k_{V2} that will make possible distinguishing between a system with leak and without leak just by observing output y. Signals that can make such distinction are called proper auxiliary signals. The best ones are called optimal proper auxiliary signals. What is the criteria for "the best" in this particular case is still an open question that has to be answered. One option might be that we want testing time to be minimal.

In terms of moles the inflows and outflows will balance at a valve. The same is not true for pressures changes since the volumes, etc., are different.

There are ways to simplify the model (1.13) even more by reducing the parameters to an independent set. Then (1.13a), (1.13b) become

$$\dot{P}_t = k_1(P_h - P_t) - k_2(P_{out} - P_{atm}) - k_3(P_t - P_{out}) + N_1\nu \quad (1.14a)$$

$$\dot{P}_h = \sigma\frac{1}{c_1P_h - c_2} - k_4(P_h - P_t) + N_2\nu \quad (1.14b)$$

The above model is nonlinear in P_h. One approach of using active fault detection of solving the above system is to observe a linearized model instead. Since we actually only use the compressor over a limited range of values of P_h, We could replace (1.14b) with the linear model

$$\dot{P}_h = \sigma(c_1 - c_2 P_h) - k_4(P_h - P_t) + N_2\nu \tag{1.15}$$

We still need to be able to estimate parameters and set the noise bound, which we did, but it is out of the scope of this thesis.

For the end of this example, if we choose not to ignore changes in temperature, based on equation (1.4) combined with equation (1.5), we will have the following model

$$\begin{aligned} \dot{P}_t &= \frac{\dot{T}_t}{T_t} P_t + T_t(k_1(P_h - P_t) - k_2(P_{out} - P_{atm}) - k_3(P_t - P_{out})) + N_1\nu && (1.16) \\ \dot{P}_h &= \sigma(c_1 - c_2 P_h) - T_h k_4(P_h - P_t) + N_2\nu && (1.17) \end{aligned}$$

Where T_t is temperature of the air in the tank and T_h is the temperature of the air at the exit of the compressor.

We repeat the above process for both, systems without leak, and systems with a small leak, hence obtaining mathematical models for both modes. The mathematical task now is to find the best way to distinguish between these two models by having control over k_{V1} and k_{V2}.

The process described in this example is necessary preparation for active fault detection. However, the focus of this thesis is on the process after the model represented by systems of differential-algebraic equations is determined. In this thesis, we are trying to find the optimal proper auxiliary controls. In the previous research on nonlinear fault detection, working with the nonlinear problem (1.14) required linearization [47]. We are going to research the ways to be able to attack nonlinear problem directly.

1.3 Objective of the Thesis

The theory for linear systems with additive noise is well developed [7], [17], [25], [18], [22]. However, a huge number of systems are of nonlinear nature. A common way to analyze those systems is using linearization and observing the newly obtained linear system. Unfortunately, that approach is not always possible or does not provide the best results. The main goal of the thesis is to analyze and develop the algorithmic approach for nonlinear systems that will allow the user to attack such problems directly. The primary focus is on the additive noise, but some research will be done for model uncertainties as well. The optimization software package SOCS developed by Boeing is used. Since we are dealing with nonlinear problems, there is no general theory how to solve every such problem, and the thesis is outlining the general approach and potential issues that might arise. Each specific nonlinear problem will require some, hopefully minimal, modifications.

Chapter 2

Active Fault Detection-Previous Results

In chapter 2, basic optimal control background will be given. The discussion about current results regarding existence of the solution and necessary and sufficient conditions will be presented. A brief literature overview of the search for global minimum and related issues will be given. The detailed derivation of Euler-Lagrange necessary conditions will be given as well. Then we will focus on the model based fault detection and how we approach the modeling of additive and model uncertainty. The presentation of active fault detection theory and results so far will follow. The basic theory of linearized approach will also be given. This will be followed by the description of the software package SOCS that is used for all examples.

2.1 Optimal Control Background

The theory of optimal control has a very long history [43], dating all the way back to Galileo and his attempts to solve geometrical problems involving maximizing area with length constraints. Legandre extended this from a solely geometric point of view to a more general framework by introducing an analytical approach that became known as *the calculus of variations.* Later, when the calculus of variations was applied to the optimization problems where the focus was on finding the best control, the field was named *optimal control theory.* More details about the history of control theory can be found in [24], [46]. While numerous books were written and research has been done in the area of theory and applications of optimal control [2], [31], [26], [35], [15], [7], thanks to the broadness of the problems, the area is far from being fully solved.

The typical and very general definition of the optimal control problem is asking us to minimize some cost function $J(t, x, u)$ whose typical presentation is:

$$J(t, x, u) = \phi(x(t_0), x(T), t_0, T) + \int_{t_0}^{T} L(x(t), u(t), t)\, dt \tag{2.1}$$

subject to differential path constraints:

$$\dot{x} = f(t, x, u) \tag{2.2}$$

together with point constraints (equalities and/or nonequalities) and algebraic path constraints. Here x is our differential variable (called the state), u is our algebraic variable (control signal) and t is time. We are trying to find the control signal u^* that will minimize cost function (2.1) subject to the constraints. Such u^* is called the optimal control signal.

There is a huge variety of problems that are varying with respect to the path and point constraints. The typical constraint differences are:

1. Initial time $t(0)$ can be free or fixed
2. Final time T can be free or fixed
3. Is there a constraint on the initial condition of the state $x(t_0)$
4. Is there a constraint on the final value of the state $x(T)$
5. Is the initial condition $x(t_0)$ known
6. Are there path constraints on the state (equalities and/or nonequalities)

7. Are there path constraints on the control (equalities and/or nonequalities)

8. Are there constraints of integral type

When approaching an optimal control problem, it is natural to ask ourselves some basic questions. Does the minimum of our optimal control problem exist, is it unique? What are the necessary, and what are the sufficient conditions that our optimal control has to satisfy to guarantee minimality? Does our solution provide a local or global minimum? The answers to the above questions, not only give us insight about the problem we are trying to solve, but also can give us an insight into how to solve the problem. Naturally, the problem specific properties of functions $\phi(x(t_0), x(T), t_0, T)$, $L(x(t), u(t), t)$ and $f(t, x, u)$ together with path and point constraints are a major part in answering those questions. There is a general theory, but the form it takes depends greatly on the specific problem.

In the discussion that follows, we should keep in mind that in the general case, all the results can be valid only locally.

The question of existence of the optimal control is quite a difficult one. There are some theoretical results for specific cases that are concerned with existence of the optimal control. In [31], the Weierstras theorem (which states that every continuous function attains its minimum on a compact set) is modified to be applicable on a general optimization problem. The result guarantees existence of the optimal control signal only if our cost functional is quasi convex and continuous, and also if our constraint set is a nonempty, closed, bounded and convex subset of a reflexive real Banach space. These requirements limit the number of problems where they can be applied on. Considering that nonlinearity in the constraints (which is the focus of this thesis) typically leads to the lack of convexity, this result would not be as useful.

In [15] the importance of compactness is stressed. To have an optimal control, it is necessary that states x satisfy some apriori bound and that time interval $[t_0, T]$ is compact. It is also important that T is large enough so that the set of feasible optimal controls is not an empty set. If the above is true, 2 classes of admissible controls that will guarantee existence are given:

1. Class of controls which satisfy Lipshitz condition
2. Class of piecewise constant controls with at most r points of discontinuity

One way to approach the question of existence will be to check whether a *feasible solution* exists. In other words, whether any control exists that satisfies the conditions of our system. This is closely related with the issue of controllability of the system. If our system is not controllable, it will be likely that no successful control will exist for some initial states. It has to be noted, that even if the system is completely controllable, it does not guarantee existence.

In the general case uniqueness can not be guaranteed. It does not exist even for a relatively simple class of problems where dynamics are autonomous linear differential equations [7]. It is not a major problem for our active fault detection applications, since in a lot of instances we do not care which of the optimal control signals we use. In special cases if we have some requirements on controls, they will be part of our constraints.

The previous discussion, while useful, does not provide insight into how to find the optimal control if it exists. The common approach in applications is to assume that a solution exists, and to try to determine necessary and sufficient conditions that the solution will have to satisfy. Lot of the inspiration for finding such conditions stem from the work of Pontryagin [42] and his maximum principle. One of the ideas is to obtain the differential of the cost function, and derive necessary equations from the fact that the differential of the cost function at any critical point has to be 0. These are the so called, first order necessary conditions. Detailed derivation of Euler-Lagrange form of these necessary conditions that we use extensively in this thesis is described in Section 2.2. There are second order necessary conditions as well. This set of conditions is concerned with second derivative of the Hamiltonian and requires that matrix to be positive semi-definite (Legandre-Clebsch condition).

There is also the well known partial differential Hamilton-Jacobi-Belman equation, whose solution with a few additional assumptions guarantees that necessary and sufficient conditions are satisfied. There is no guarantee that there is a solution, but if we find one it will prove an existence of optimal control signal. Solving this problem is done computationally and can be a very challenging task in the computer algorithm sense. It was suggested [2] to use HJB equation as a test for the candidate for optimality.

Sufficient conditions are guaranteeing us whether our solution is really an optimal control. There are a few theoretical results, but they are usually applicable only to a limited number of problems. For example, Mangasarian's sufficient condition requires convexity and fixed time among other constraints. For a nice overview of the sufficient conditions consult the survey [44]. It is also worth noting the analysis of the second order sufficient conditions using a Riccati approach in [36]. This work is significant for our thesis since it does not require the final time to be fixed, which might be exploited in our approach to model uncertainty.

While all the results from above can help us find candidates for optimal control signals and even check whether they provide a minimum of the cost function (instead of a maximum or saddle point), in most cases they are not guaranteeing that the found minimum is global. Since today, most (maybe all) of the complex optimal control problems are solved using computer programs, the search for a global minimum is a part of the area of *global optimization.* There is extensive research about the methods that would help find global minimum by applying a proper algorithmic approach [40], [45],[32],[23].

2.2 Necessary Conditions

One of the well researched areas of optimal control theory [2], [33], [35] is dealing with the model of some physical system represented as a system of first order differential equations

$$\dot{x} = f(x, u, t) \tag{2.3}$$

that has performance index (also called a cost function) associated with it of the form

$$J(t_0) = \phi(x(t_0), x(T), t_0, T) + \int_{t_0}^{T} L(x(t), u(t), t)\ dt \tag{2.4}$$

where x is our differential variable (called the state), u is our algebraic variable (control signal) and t is time. The optimal control problem is finding the control signal u^* on a given time interval $[t_0, T]$ that will minimize cost function (2.4). Such u^* is called the optimal control signal. In our problem, t_0 and T will be fixed but in general they may also be variables. There may be additional constraints which we will discuss later.

There is a well known approach from optimal control theory that gives necessary conditions for this problem using calculus of variations and Lagrange multipliers. Since the procedure will be used many times in this thesis, we will briefly present that approach.

First the augmented cost function $\bar{J}$ is created. It combines our system of differential equations (2.3) with our initial cost function (2.4) using Lagrange multiplier $\lambda(t)$. Notice that $\lambda(t)$, $x(t)$ and $u(t)$ are in general column vectors.

$$\bar{J} = \phi(x(t_0), x(T), t_0, T) + \int_{t_0}^{T} L(x(t), u(t), t) + \lambda^T(t)(f(x, u, t) - \dot{x})\ dt \tag{2.5}$$

For simplicity of the final result, we introduce the Hamiltonian function

$$H(x, u, t) = L(x(t), u(t), t) + \lambda^T(t) f(x, u, t) \tag{2.6}$$

and substituting (2.6) in (2.5) we obtain the following form:

$$\bar{J} = \phi(x(t_0), x(T), t_0, T) + \int_{t_0}^{T} H(x, u, t) - \lambda^T(t)\dot{x}\ dt \tag{2.7}$$

The necessary condition for minimum of cost function J requires that increment $dJ = 0$ along the constraints (2.3). Lagrange's theory states that $dJ = 0$ along constraints (2.3) is achieved when $d\bar{J} = 0$.

Note that differentials dx, du and $d\lambda$ are not independent of differential dt. As usually done in the calculus of variations [35], we will define variations (increments) δx,δu and $\delta\lambda$ as the incremental changes in corresponding variables when time t is held fixed, hence variations are independent of dt.

Now we want to express $d\bar{J}$ as a function of variations in all variables, δx,δu and $\delta\lambda$. For the next step, we need Leibniz's rule for functionals that states that if

$$K(x) = \int_{t_0}^{T} g(x(t), t)\ dt \tag{2.8}$$

and T and t_0 are allowed to vary then

$$dK(x) = g(x(T), T)dT - g(x(t_0), t_0)dt_0 + \int_{t_0}^{T} g_x^T(x(t), t)\delta x\ dt \tag{2.9}$$

Using Leibniz's rule several times on (2.7), we obtain an expression for $d\bar{J}$

$$\begin{aligned} d\bar{J} = \phi_x^T dx|_{t=t_0} \quad &+ \quad \phi_x^T dx|_{t=T} + (\phi_t + H - \lambda^T\dot{x})\ dt|_{t=T} + (\phi_t - H + \lambda^T\dot{x})\ dt|_{t=t_0} \\ &+ \quad \int_{t_0}^{T} [H_x^T\delta x + H_u^T\delta u - \lambda^T\delta\dot{x} + (H_\lambda - \dot{x})^T\delta\lambda]\ dt \end{aligned} \tag{2.10}$$

Using integration by parts, we can eliminate $\delta\dot{x}$ since

$$\int_{t_0}^{T} \lambda^T\delta\dot{x}\ dt = \lambda^T\delta x|_{t=T} - \lambda^T\delta x|_{t=t_0} - \int_{t_0}^{T} \dot{\lambda}^T\delta x\ dt \tag{2.11}$$

Using the calculus of variations [35], for small variations, the terms $\delta x|_{t=T}$ and $\delta x|_{t=t_0}$ from (2.11) can be expressed as

$$\delta x|_{t=T} \quad = \quad dx|_{t=T} - \dot{x}(T)\ dt|_{t=T} \tag{2.12a}$$

$$\delta x|_{t=t_0} \quad = \quad dx|_{t=t_0} - \dot{x}(t_0)\ dt|_{t=t_0} \tag{2.12b}$$

Combining the above equations we get the final general form for the increment $d\bar{J}$:

$$\begin{aligned} d\bar{J} \quad = \quad & (\phi_x + \lambda)^T dx|_{t=t_0} + (\phi_x - \lambda)^T dx|_{t=T} + (\phi_t - H)dt|_{t=t_0} \\ & + (\phi_t + H)dt|_{t=T} + \int_{t_0}^{T} [(H_x + \dot{\lambda})^T\delta x + H_u^T\delta u + (H_\lambda - \dot{x})^T\delta\lambda]\ dt \end{aligned} \tag{2.13}$$

For $d\bar{J} = 0$ to be satisfied, factors with independent increments have to be 0. Observing the integrand, we obtain the following equations that have to be satisfied:

$$\dot{x} = H_\lambda \tag{2.14a}$$

$$\dot{\lambda} = -H_x \tag{2.14b}$$

$$0 = H_u \tag{2.14c}$$

Regarding the rest of $d\bar{J}$, some terms can be eliminated if starting and ending time or state are fixed. For this thesis, t_0 and T are fixed, hence corresponding increments are 0. This gives us the boundary conditions that have to be satisfied.

$$(\phi_x + \lambda)^T dx|_{t=t_0} = 0 \tag{2.15a}$$

$$(\phi_x - \lambda)^T dx|_{t=T} = 0 \tag{2.15b}$$

Depending on the problem, attention has to be given whether values of states x at times t_0 or T are independent.

2.3 Modeling and Basic Principles of Active Fault Detection

When applying active fault detection we are observing the system using a white box presentation. In other words, we are relying on the knowledge that we know a mathematical model presentation of the system. Typically, we obtain such model in the following ways

1. Experimentally, by observing input-output data and observable states from the physical system

2. Theoretically, by deriving input-output relationships from physical laws that govern our system

3. Combination of the previous two

Standard components of our mathematical model are

1. System of differential equations describing dynamics of the states of the system
2. System of algebraic equations describing what our outputs will be
3. System of equations or nonequalities that account for physical constraints of the system
4. Noise components that will accommodate for errors of our model or the uncertainties of the system itself

Since most of the work done so far in the literature is focused on the linear models, for the rest of this chapter we will focus on linear models as well. Thus we assume that observed system can be modeled by the following system of differential-algebraic equations (DAE)

$$\dot{x} = (A + M\Delta G)x + (B + M\Delta H)v + \overline{B}\nu \tag{2.16a}$$

$$y = (C + N\Delta G)x + (D + N\Delta H)v + \overline{D}\nu \tag{2.16b}$$

Where

1. $x \in R^n$ represents n states of the system
2. $y \in R^m$ represents m observed outputs
3. $v \in R^l$ represents l control test signals
4. $\nu \in R^a$ represent additive noise component in the system
5. Matrices $\overline{B}$ and $\overline{D}$ are weights on additive noise
6. Matrices M, N, G, H and Δ are used to represent multiplicative (model) uncertainties

Usually $n > m$ and l is close to 1. Since the presence of model uncertainties has unique additional challenges, we are first going to present existing relevant results for the case when only additive uncertainty is present. Notice that all we have to do is set $\Delta = 0$. We are going to use the following subsection to also present basic principles of active fault detection.

2.3.1 Active Fault Detection on Linear Systems with Additive Uncertainty

To apply this method, we have to know not only the model of the fully operational system, but also the model of the system when some specific fault is present. The model for faulty system can be obtained in the same ways as the model of the fully functional system. Though results and procedure exist [7] for multiple faults, in this thesis we will focus only on detection of a single fault. By setting $\Delta = 0$ in equations (2.16), the system of DAE for both models is

$$\dot{x_0} = A_0 x_0 + B_0 v + M_0 \nu_0 \tag{2.17a}$$

$$\dot{x_1} = A_1 x_1 + B_1 v + M_1 \nu_1 \tag{2.17b}$$

$$y_0 = C_0 x_0 + N_0 \nu_0 \tag{2.17c}$$

$$y_1 = C_1 x_1 + N_1 \nu_1 \tag{2.17d}$$

Index 0 represents model of fully functional system, while index 1 represents model for system where fault is present.

Notice the discrepancy between notation for equations (2.16) and (2.17). In particular that matrix $\overline{B}$ from (2.16) is now matrix M and $\overline{D}$ from (2.16) is now matrix N. The reason for discrepancy is that we are following notation in the literature [7], which is not totally consistent. Only direct feed through systems where auxiliary signal v does not influence the output y directly are observed. Hence matrix $D = 0$.

If any amount of uncertainty is allowed, in the general case there is not much we can do. Fortunately, in most cases we can determine some limit on the expected amount of uncertainty in the system, usually using statistics. So let's define a bound on the amount of uncertainty $S_i, i = 0, 1$ as:

$$S_i = (x_i(t_0) - x_{i0})^T P_i(x_i(t_0) - x_{i0}) + \int_0^s \nu_i^T \Gamma_i \nu_i \, dt < \rho, \forall s \in [t_0, T] \qquad (2.18)$$

where $(x_i(t_0) - x_{i0})^T P_i(x_i(t_0) - x_{i0})$ represents uncertainty of the initial condition and $\Gamma > 0$ is our weight matrix on noise components. For the additive noise only case the following simplifications are valid

1. $\Gamma_i = I$, where I is identity matrix of proper dimensions
2. Upper bound of the integral in (2.18) is at $s = T$
3. Bound $\rho = 1$

We will discuss the model uncertainty case in the Section 2.3.2. While ρ is a predetermined bound and depends on the case by case basis, we can manipulate that value for our model by modifying matrices M_i and N_i. Hence, without loss of generality, for the rest of this subsection we take $\rho = 1$.

It is a reasonable assumption that outputs y and inputs v to the system are known. Based on that knowledge, we would like to be able to determine whether our system operates in fully operational mode or in the faulty mode, i.e. whether available input-output data is achievable by system of DAE equations with model index 0 or 1, while still satisfying noise bounds S_i, for $i = 0, 1$. If we are applying some arbitrary input v, it is quite possible that our input-output pair could correspond to both systems, which would make decision impossible. The basic idea of active fault detection is to design input v that would guarantee that just by observing input-output set we will be able to conclude which model corresponds to the given system. Such signal v is called **proper auxiliary signal** v. This feature is represented on the Figure 2.1 where, while two input-output sets can overlap in general, after applying proper signal

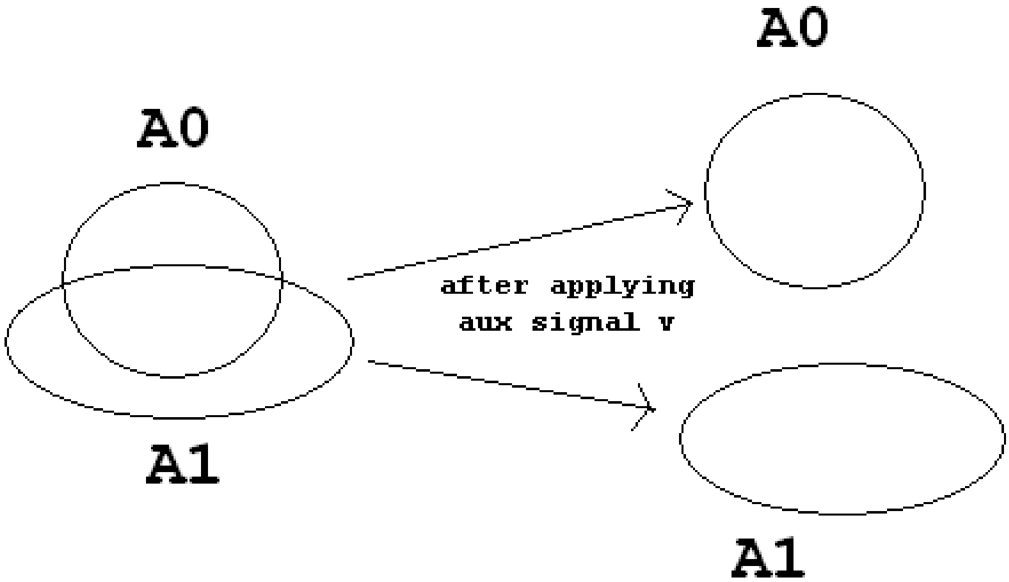

Figure 2.1: Input-output set separation by proper signal v

v those sets are separated. Thus, for a given proper v, if noise bounds $S_i, i = 0, 1$ are satisfied, outputs from two models have to be different. Another way to express this is that for a given proper v, if two models have the same output, at least one of the noise bounds $S_i, i = 0, 1$ has to be broken. Equivalent mathematical formulation for the later logic is

$$\dot{x_0} = A_0 x_0 + B_0 v + M_0 \nu_0 \tag{2.19a}$$

$$\dot{x_1} = A_1 x_1 + B_1 v + M_1 \nu_1 \tag{2.19b}$$

$$0 = C_0 x_0 + N_0 \nu_0 - C_1 x_1 - N_1 \nu_1 \tag{2.19c}$$

$$\phi(v) = \inf_{x_i, \nu_i, v} \max\{S_0, S_1\} \geq \rho \tag{2.19d}$$

We not only want to find proper signal v, but we would like to find "the best" proper signal v^* of all proper signals v. Such signal v^* is called the **optimal proper auxiliary signal** v^*. There can be many criteria for what is "the best" proper signal. The choice depends on the specific problem. Two most common criteria for optimal proper auxiliary signal v^* are

1. It is the proper signal v with the smallest L^2 norm
2. It is the proper signal v that disturbs the system the least

The disturbance of the system can be measured by introducing a new variable ψ whose behavior is described by

$$\dot{\psi} = A_0\psi + B_0v, \psi(t_0) = 0 \tag{2.20}$$

Notice that in this thesis dynamics are the same as for working model without additive noise. That does not have to be the case.

Using this definition of disturbance, the measure of our proper auxiliary signal v can be expressed as

$$\delta^2(v) = \psi(T)^T W\psi(T) + \int_{t_0}^{T} |v|^2 + \psi^T U\psi \; dt \tag{2.21}$$

where U and W are positive semidefinite matrices. Notice that if $U = W = 0$, measure becomes just a simple L^2 norm of auxiliary signal v. Note that $\delta(v)$ is an inner product norm on v.

Hence our optimization problem can be defined as

$$\min_v \delta^2(v)$$

subject to

$$\dot{x_0} = A_0x_0 + B_0v + M_0\nu_0 \tag{2.22a}$$

$$\dot{x_1} = A_1x_1 + B_1v + M_1\nu_1 \tag{2.22b}$$

$$\dot{\psi} = A_0\psi + B_0v \tag{2.22c}$$

$$0 = C_0x_0 + N_0\nu_0 - C_1x_1 + N_1\nu_1 \tag{2.22d}$$

$$\psi(t_0) = 0 \tag{2.22e}$$

$$\phi(v) = \inf_{x_i,\nu_i,v} \max\{S_0, S_1\} \geq \rho \tag{2.22f}$$

where

$$S_i = (x_i(t_0) - x_{i0})^T P_i (x_i(t_0) - x_{i0}) + \int_{t_0}^{T} |\nu_i|^2 \ dt \tag{2.23}$$

There are 2 approaches to solving this problem. The first approach is solving these equations by using control theory. For linear models, an elegant theory utilizing Riccati equations is developed in [7]. However, our interest is nonlinear problems, so in this thesis we will focus on the second approach that uses the **direct optimization formulation**. The idea is that if we can rewrite our problem as a standard optimal control problem, we will be able to use existing powerful optimal control solvers. The issue is that equation (2.22f) can not be applied in a given form in existing solvers.

In research prior to this thesis, the crucial step in conversion of equation (2.22f) into a more acceptable form was the assumption that we can exchange the order of the infimum and maximum. It was proved that exchange is justifiable for linear problems, however it does not necessarily hold for nonlinear problems.

For the conversion we will also need the following fact that for some parameter β, $0 \le \beta \le 1$

$$\max\{S_0, S_1\} = \max_{\beta} \left(\beta S_0 + (1-\beta) S_1\right) \tag{2.24}$$

After exchanging the order of the infimum and maximum, and substituting (2.24) and (2.23) in (2.22f) we obtain the equation

$$\phi(v) = \max_{\beta \in [0,1]} \inf_{x_i, \nu_i, v} \{Z(T)\} \ge \rho \tag{2.25}$$

where new variable $Z(t)$ is defined by

$$\dot{Z}(t) = \beta \left|\nu_0\right|^2 + (1-\beta) \left|\nu_1\right|^2 \tag{2.26}$$

$$\begin{aligned} Z(t_0) &= \beta (x_0(t_0) - x_{00})^T P_0 (x_0(t_0) - x_{00}) \\ &\quad + (1-\beta)(x_1(t_0) - x_{10})^T P_1 (x_1(t_0) - x_{10}) \end{aligned} \tag{2.27}$$

We introduced a new variable $Z(t)$, which increased the complexity of the problem by adding one more differential equation and boundary condition (typically not a significant increase), however, the advantage is that now just by observing value $Z(T)$ we can conclude whether one of the noise bounds is broken or not.

If $\inf_{x_i,\nu_i} Z(T) \geq \rho$ **for some** $\beta \in [0,1]$ **then** $\phi(v) \geq \rho$**.**

Our new presentation of the problem is

$$\min_v \delta^2(v)$$

subject to

$$\begin{aligned}
\Theta &= \min_{x_i,\nu_i} Z(T) && (2.28a)\\
\dot{x_0} &= A_0 x_0 + B_0 v + M_0 \nu_0 && (2.28b)\\
\dot{x_1} &= A_1 x_1 + B_1 v + M_1 \nu_1 && (2.28c)\\
\dot{Z}(t) &= \beta \left|\nu_0\right|^2 + (1-\beta)\left|\nu_1\right|^2 && (2.28d)\\
0 &= C_0 x_0 + N_0 \nu_0 - C_1 x_1 + N_1 \nu_1 && (2.28e)
\end{aligned}$$

with additional constraints

$$\begin{aligned}
\dot{\psi} &= A_0 \psi + B_0 v && (2.29a)\\
Z(t_0) &= \beta (x_0(t_0) - x_{00})^T P_0 (x_0(t_0) - x_{00}) \\
&\quad + (1-\beta)(x_1(t_0) - x_{10})^T P_1 (x_1(t_0) - x_{10}) && (2.29b)\\
\psi(t_0) &= 0 && (2.29c)\\
0 &\leq \beta \leq 1 && (2.29d)\\
\Theta &\geq \rho && (2.29e)
\end{aligned}$$

We can observe that (2.28) form another optimization problem on their own, usually referred to as the **inner optimization problem**. Let's assume that all of the N_i matrices has a full row rank. Then we can use the system of algebraic equations (2.28e) to solve for some components of the noise ν_i. Then we can replace those components of ν_i in other equations thus not only reducing the number of variables, but also eliminating the additional need for the system of algebraic equations themselves. Optimal control problem consisting only of modified differential constraints (2.28b),(2.28c) and (2.28d) can now be transformed into a system of DAE as shown in Section 2.1. More details can be found in chapter 4 of [7].

So finally, our problem is defined as

$$\min_{v} \delta^2(v)$$

subject to:

System of DAE from inner optimization problem solution and

$$\dot{\psi} = A_0\psi + B_0 v \tag{2.30a}$$

$$Z(t_0) = \beta(x_0(t_0) - x_{00})^T P_0(x_0(t_0) - x_{00}) + (1-\beta)(x_1(t_0) - x_{10})^T P_1(x_1(t_0) - x_{10}) \tag{2.30b}$$

$$\psi(t_0) = 0 \tag{2.30c}$$

$$0 \leq \beta \leq 1 \tag{2.30d}$$

$$\Theta \geq \rho \tag{2.30e}$$

This formulation represents a standard optimal control problem which can be presented to some of the existing optimal control solvers.

The theory for linear systems with additive noise is well developed and numerous examples and extensions are done [7],[25],[18],[22],[17]. It was shown that v^* exists. The further steps which describe how to implement optimal proper signal v^* to decide which model is currently present using the **hyperplane test** are also examined in detail. To be able to use the hyperplane test, it is necessary that our input-output sets be convex. Since that is not the case for nonlinear systems, the hyperplane test cannot be used in general. The standard test has to be used instead. Later we will have to see for a calculated auxiliary signal and observed inputs and outputs, what is the noise required in the system. The one that breaks the bound loses.

Certain extensions are done as well in the literature such as

1. Systems with additional input signals beside auxiliary signal v
2. Systems with constraints on the auxiliary signal v
3. Systems discretized in time
4. Systems with delay [18]
5. Having more than one possible fault
6. Early detection of the fault using hyperplane test

2.3.2 Formulation of the Linear Systems with Model and Additive Uncertainty

While (2.16) can be used to describe this case, it would be convenient if we can rewrite our problem in the form that resembles system of equations used in Section 2.3.1. Beside additive noise bound (2.18), we will also have a bound on model uncertainty of the form $\overline{\sigma}(\Delta(t)) \leq B_m$. Without loss of generality, we can assume $B_m = 1$ since we can manipulate with matrices M, N, G and H. As in Section 2.3.1 we can assume that the bound on additive uncertainty is $\rho = 1$. As shown in [7], instead of original system:

$$\dot{x} = (A + M\Delta G)x + (B + M\Delta H)v + \overline{B}\nu \tag{2.31a}$$

$$y = (C + N\Delta G)x + (D + N\Delta H)v + \overline{D}\nu \tag{2.31b}$$

$$\overline{\sigma}(\Delta(t)) \leq 1 \tag{2.31c}$$

$$S = (x(t_0) - x_0)^T P(x(t_0) - x_0) + \int_0^s \nu^T \Gamma \nu \, dt < 1 \tag{2.31d}$$

$\forall s \in [t_0, T]$ where $\overline{\sigma}(\Delta(t))$ is the largest singular value of Δ, we can use the modified system:

$$\dot{x} = Ax + Bv + M\mu + \overline{B}\nu \tag{2.32a}$$

$$y = Cx + Dv + N\mu + \overline{D}\nu \tag{2.32b}$$

$$\xi = Gx + Hv \tag{2.32c}$$

with $\mu = \Delta\xi$.

To assure that model noise bound $\overline{\sigma}(\Delta(t)) \leq 1$ is not broken, we have the condition that

$$|\mu| - |\xi| \leq 0, \;\; \forall t \in [0, T] \tag{2.33}$$

To be able to fully represent this problem in the additive noise form, we can rewrite (2.33) as

$$\int_0^s |\mu|^2 - |\xi|^2 \;\, dt < 0, \;\; \forall s \in [0, T] \tag{2.34}$$

Notice, that the new definition allows for more noise, since it allows $|\mu| \geq |\xi|$ for some periods of time.

Adding the bounds (2.31d) and (2.34) the overall noise bound is:

$$S = (x(0) - x_0)^T P_0^{-1}(x(0) - x_0) + \int_0^s |\nu|^2 + |\mu|^2 - |\xi|^2 \;\, dt < 1, \forall s \in [0, T] \tag{2.35}$$

Notice that here we allow even more noise in. This way we introduce more conservatism in our solution. Auxiliary signal v obtained this way will certainly be proper. However, since we require v to handle more noise than necessary, it is reasonable to expect that it is going to be suboptimal, i.e. there should be another proper signal $v^* \neq v$ whose cost function $\delta^2(v^*)$ is smaller than $\delta^2(v)$.

We would like to point out that while auxiliary signal v obtained this way will be proper, it will be suboptimal.

Equation (2.35)can be rewritten as

$$S = (x(0) - x_0)^T P_0^{-1}(x(0) - x_0) + \int_0^s \varphi^T \Gamma \varphi \; dt < 1, \forall s \in [0, T] \tag{2.36}$$

with $\varphi = \begin{bmatrix} \nu \\ \mu \\ \xi \end{bmatrix}$ and $\Gamma = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & -I \end{bmatrix}$.

Finally, observe that the form of the system of differential-algebraic equations (2.32) together with condition that noise bound is not broken (2.36) that describe the case where model uncertainty is present, is very similar to the presentation of linear models when only the additive noise is present.

Two big differences between these two problems being

- Matrix Γ is positive definite for additive noise case, while it does not have to be positive definite for the model uncertainty case
- As a consequence, for additive noise case we needed to check whether bound (2.36) is broken for $s = T$ only, while for model uncertainty case we need to make sure that noise bound is not broken $\forall s \in [0, T]$

Applications of linear problems with model uncertainty in active fault detection were done to certain extent [16],[7], but not for the direct optimization approach. This thesis will present the first direct optimization solution of the model uncertainty problem.

2.3.3 Nonlinear Results and Approach

In the research so far, the common approach to nonlinear problems was to apply linearization and solve the newly obtained problem. If we know one solution ζ_0 of the nonlinear differential equation

$$\dot{x} = F(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \dots \\ f_n(x) \end{bmatrix} \tag{2.37}$$

we have

$$\dot{x} = F(\zeta_0) + J_F(\zeta_0)(x - \zeta_0) + O(|x - \zeta_0|^2) \tag{2.38}$$

and we can linearize this equation as

$$\dot{x} = F(\zeta_0) + J_F(\zeta_0)(x - \zeta_0) \tag{2.39}$$

where $J_F(x)$ is the Jacobian matrix $J_F(x) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \cdots & \cdots & \cdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}$. In [25], under future work and conclusions, 3 specific cases of nonlinearities were discussed that might be solved by directly using existing algorithms developed for linear problems.

1. **Small nonlinearities**

   If our differential equation is given as

   $$\dot{x} = Ax + g(x,t) + Bv + M\nu \tag{2.40}$$

   where the nonlinearity $g(x,t)$ is bounded by $\|g(x,t)\| < \epsilon$ and the original noise bound is $\|\nu\| < B$, we can translate this problem into a linear one with the new noise bound $\|\nu\| < B + \epsilon t_f$. The auxiliary signal v obtained this way will be conservative, in the sense that it will be able to handle even more noise than necessary. Hence our solution will be suboptimal.

2. **Nonlinear in control**

   For this case, nonlinearity in the system is introduced only through nonlinearity in v as

   $$\dot{x} = Ax + g(v) + M\nu \tag{2.41}$$

   This type of nonlinearity does not require any special modifications of the equations. However, depending on the specifics of $g(v)$, we could have multiple optimal v and the quality of the solution will depend on the good initial guess.

3. **Coefficient matrices dependent on the detection signal**

   The problem formulation is of the type

$$\dot{x} = A(v)x + B(v)v + M(v)\nu \quad (2.42a)$$

$$y = C(v)x + N(v)\nu \quad (2.42b)$$

Recently [1],[47], the theoretical and practical approach for examining under which circumstances linearization is justified has been researched. While [47] is tackling nonlinear problems as well, the focus of that thesis and our work is quite different. In [47] the focus is on

- Theory to support use of linearization for nonlinear problems
- Apply linear algorithms using SIMULINK models to solve the problem
- Additive uncertainty cases

while the focus of this thesis is on

- Directly attack nonlinear problem
- Set up and analyze systems of nonlinear equations
- Solve them numerically using SOCS software package
- Additive and model uncertainty cases

2.3.4 Optimal Control Solver SOCS and Other Software

For this thesis, we are mostly using the Sparse Optimal Control Software (SOCS) package developed by the Boeing corporation [6],[3] and further information can be found on their website http://www.boeing.com/phantom/socs.

This software package is a collection of Fortran 77 routines and is designed with optimal control problems in mind (also it is designed for nonlinear optimization and parameter estimation problems, but we are not focusing on that). SOCS uses a direct transcription method (collocation method) to convert the continuous system of our differential-algebraic equations into its discretized approximation. That way, SOCS deals with a sparse, finite dimensional nonlinear programming problem and solves it as such.

The consequence of direct transcription approach on our problems is that typically the size of the nonlinear programming problem tends to be quite large. That is why it is of great importance to reduce the number of variables and algebraic equations as much as possible to make the problem easier for SOCS to solve. This is done by simply expressing one variable (usually the noise component in our cases) in terms of the others and replacing it in all other equations.

Among the many features of this powerful package, the following were very important for our thesis:

1. **SOCS allows us to import initial guess to our driver.**

   Since for nonlinear problems, multiple minimums and potential maximums are often occurring, the possibility to control the initial guess is of utmost importance to control what kind of solution (if any) we will get.

2. **Length of the test interval does not have to be fixed.**

   This feature becomes extremely handy for model uncertainty cases.

3. **Allows multiple phases.**

   This also might be useful for model uncertainty case where we want conditions for a certain time interval to be different than for the remainder of the time interval.

4. **Allows outside routines to define desired functions.**

   One of the ideas for the approach for solving model uncertainty is counting on this feature.

5. **SOCS works with both, equality and inequality path constraints.**
   Both types of path or variables constraints are often found in optimal control problems.

6. **We have control of which numerical discretization method to use.**
   We encountered examples where just switching from one discretization method to another makes the difference between SOCS not being able to solve the problem to finding correct solution.

7. **Flexible initial grid size.**
   Choosing too coarse an initial grid might increase the time required to get the solution, while starting with too fine grid might make solution impossible. Thus being able to fidget with this parameter was convenient. Also, when we were importing an initial guess, we knew exactly what value of the initial grid we want to have.

8. **We have control over the permitted numerical error.**
   This is a standard and always useful feature for any software.

Beside numerous drivers written for SOCS, in this thesis we used MATLAB (http://www.mathworks.com) and MAPLE (http://www.maplesoft.com) software packages mostly for equation preparation for SOCS and for creation of the figures. Our code was verified by comparing the solution of some simpler linear problems to solutions obtained by codes written in Scilab [8], [7] and run by K. Sweetingham.

Chapter 3

Active Fault Detection for Nonlinear Systems

In this chapter, the nonlinear problem will be defined and the novel approach using the p-norm will be introduced. Then we will derive necessary conditions for nonlinear systems and linear systems with additive noise. The analysis of benefits and weaknesses of this approach compared to the standard β approach will be given as well as a few examples. Then we will focus on the linear problems with model uncertainty, where we will present an algorithm for solving those problems using the software package SOCS (Section 2.3.4). This will be the first direct optimization formulation of the model uncertainty problem.

3.1 Definition of the Problem

In this section, we are building on previous results presented in Chapter 2 applied to nonlinear systems. As before, we are observing the system using white box presentation and we focus on the detection of specific faults. Models of nonfaulty and faulty systems can be expressed as:

$$\dot{x_0} = f_0(x_0, u, \nu_0, v) \quad (3.1a)$$

$$\dot{x_1} = f_1(x_1, u, \nu_1, v) \quad (3.1b)$$

$$y_0 = g_0(x_0, u, \nu_0, v) \quad (3.1c)$$

$$y_1 = g_1(x_1, u, \nu_1, v) \quad (3.1d)$$

Where

1. $x_0, x_1 \in R^n$ represents n states of the system
2. $y \in R^m$ represents m observed outputs
3. $u \in R^k$ represents additional k input signals
4. $v \in R^l$ represents l auxiliary control signals
5. $\nu_0, \nu_1 \in R^a$ represent noise in the system

Usually $n > m$ and k and l are close to 1.

For the rest of the thesis we will ignore additional inputs u and assume that the starting time of the process is $t_0 = 0$.

Define the bound on the amount of uncertainty $S_i, i = 0, 1$ as:

$$S_i = (x_i(0) - x_0)^T P_i(x_i(0) - x_0) + \int_0^s \nu_i^T \Gamma \nu_i \; dt < \rho, \; \forall s \in [0, T] \quad (3.2)$$

where $(x_i(0) - x_0)^T P_i(x_i(0) - x_0)$ represents uncertainty of the initial condition and Γ is our weight matrix on noise components.

If both failed and unfailed model, can produce the same outputs and satisfy uncertainty constraint, we will not be able to detect failure. Following the basic principle of active fault detection, our goal is to design auxiliary signal v such that in order to satisfy (3.1a), (3.1b) and have the same output, one or both uncertainty constraints (3.2) have to be broken. Such an auxiliary signal v is called *proper*.

We would like to find the *optimal proper auxiliary signal* v^* which is "the best" of all proper signals in some sense. It should be important to notice that while we are striving to find optimal v^*, often we will end up with suboptimal, but still proper signal v. One of the criteria to choose v^* [7] is to find proper signal v with the smallest L^2 norm and/or which causes the least disturbance to the states of the system ψ.

The disturbance can be described as

$$\dot{\psi} = f(\psi, v), \;\; \psi(0) = 0 \tag{3.3}$$

Usually $f(\psi, v) = f_0(\psi, 0, 0, v)$, i.e. the same as function f_0 when there is no noise and additional inputs u.

The measure of our auxiliary signal v thus becomes

$$\delta^2(v) = \psi(T)^T W \psi(T) + \int_0^T |v|^2 + \psi^T U \psi \; dt \tag{3.4}$$

where U and W are positive semidefinite matrices and δ is a norm on L^2. Notice that if $U = W = 0$, $\delta(v)$ becomes just a simple L^2 norm of auxiliary signal v.

A proper signal will cause one or both uncertainty constraints to be broken.

$$\phi(v, s) = \inf_{x_i, \nu_i, y, u, s \in [0,T]} \max\{S_0, S_1\} \geq \rho \tag{3.5}$$

Existing results, applicable for linear models only, are using the facts that for $0 \leq \beta \leq 1$:

1. $\max\{S_0, S_1\} = \max_\beta(\beta S_0 + (1 - \beta) S_1)$

2. Infimum and maximum can exchange order

3. For additive noise only, time parameter $s = T$

Using the above assumptions, a detailed theory was developed as presented in chapter 2.

However, for nonlinear models, the infimum and maximum can not exchange order. As a way to bypass that problem, in this thesis a suboptimal approach is proposed.

3.1.1 p-norm Approach

Using the fact that p-norm approaches the max norm as $p \to \infty$, we are proposing approximation of max norm in equation (3.5) with p-norm:

$$\phi(v, s) = \inf_{x_i, \nu_i, y, u, s \in [0,T]} \sqrt[p]{S_0^p + S_1^p} \geq \rho \tag{3.6}$$

with relative error bound $\sqrt[p]{2}$.

Proof: Without loss of generality, assume that $S_0 > S_1$. Then:

$$\frac{\sqrt[p]{S_0^p + S_1^p}}{S_0} \leq \frac{\sqrt[p]{S_0^p + S_0^p}}{S_0} \leq \sqrt[p]{2} \;\; QED$$

Since $S_0 > 0$ and $S_1 > 0$, we have shown that max norm and p-norm bounds are intertwined

$$\max\{S_0, S_1\} \leq \sqrt[p]{S_0^p + S_1^p} \leq \sqrt[p]{2} \max\{S_0, S_1\} \tag{3.7}$$

Hence, to assure that the found auxiliary signal is proper, noise bound ρ has to be multiplied by $\sqrt[p]{2}$. Note that for numerical stability purposes, it is more convenient to express the noise bound as

$$2\rho^p \leq \inf_{x_i, \nu_i, y, u, s \in [0,T]} S_0^p + S_1^p \tag{3.8}$$

instead of

$$\sqrt[p]{2}\rho \leq \inf_{x_i, \nu_i, y, u, s \in [0,T]} \sqrt[p]{S_0^p + S_1^p} \tag{3.9}$$

The relation between the p-norm and the max norm is presented in figure 3.1. As can be seen, if we keep the same noise bound for p-norm (solid lines), there is a pair of S_0 and S_1 that, though the pair breaks the p-norm, still satisfies the max norm. Hence we need an adjustment for the bound as the theorem stated. When we apply the theorem suggested adjustment (dotted lines), we can see that breaking of p-norm bond guarantees that max norm is broken as well, thus assuring properness of found signal v. We can also notice that we are letting more noise in the system which will result in a suboptimal proper signal v being found this way. From the figure 3.1 we can also notice the fact that as we increase value p, p-norm approaches max norm.

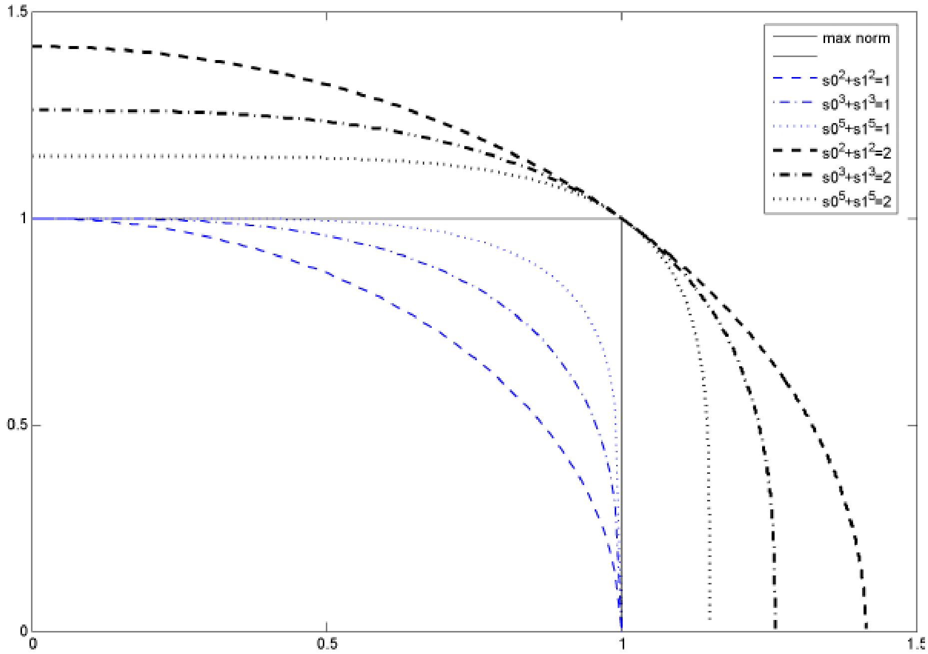

Figure 3.1: Comparison of p-norm and max norm

Combining all of the above, our problem is to

$$\text{minimize } \psi(T)^T W \psi(T) + \int_0^T |v|^2 + \psi^T U \psi \; dt$$

subject to

$$\dot{x_0} = f_0(x_0, u, \nu_0, v) \tag{3.10a}$$

$$\dot{x_1} = f_1(x_1, u, \nu_1, v) \tag{3.10b}$$

$$\dot{\psi} = f(\psi, v), \;\; \psi(0) = 0 \tag{3.10c}$$

$$0 = g_0(x_0, u, \nu_0, v) - g_1(x_1, u, \nu_1, v) \tag{3.10d}$$

$$2\rho^p \leq \inf_{x_i, \nu_i, y, u, s \in [0,T]} S_0^p + S_1^p \tag{3.10e}$$

As a consequence of this suboptimal approach, we would expect that for linear problems the p-norm method would produce a larger proper auxiliary signal and that, as p increases, the computed proper auxiliary signals are approaching the optimal one.

It is always a good idea to check an approach on a problem with a known result. We are going to use the following example:

p-norm approach applied on linear problem

This example is inspired by Example 4.2.2 from [7] describing equalized and linearized model of a single engine F-16 aircraft [50].

Objective is to minimize $\int_0^1 |v|^2 \, dt$ subject to

$$\dot{x_0} = \begin{bmatrix} -0.1689 & 0.0759 & -0.9952 \\ -26.859 & -2.5472 & 0.0689 \\ 9.3603 & -0.1773 & -2.4792 \end{bmatrix} x_0 + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} v + \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \nu_0 \tag{3.11a}$$

$$\dot{x_1} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} x_1 + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} v + 2 \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \nu_1 \tag{3.11b}$$

$$0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.9971 & 0.0755 \end{bmatrix} (x_0 - x_1) + \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} (\nu_0 - 2\nu_1) \tag{3.11c}$$

$$\sqrt[p]{2} \leq \inf_{x_i, \nu_i} S_0^p + S_1^p \tag{3.11d}$$

where $S_i = \int_0^1 |\nu_i|^2 \, dt$, $\rho = 1$ and p represents chosen p-norm. The problem is slightly modified to allow 2 times more noise for the faulty model. Before writing driver for SOCS, we simplified the above system of differential-algebraic equations by using algebraic equations (3.11c) to eliminate some noise variables and convert the problem to a system of differential equations. The found auxiliary signals are in Figure 3.2. The calculated cost functions and noise bounds are in table 3.1. The code that calculates noise bound is one of the contributions of this thesis, and is described in details in Section 3.4. The results are as expected. As p grows, the auxiliary signals are getting closer to the optimal auxiliary signal. We also notice that our suboptimal auxiliary signal obtained by p-norm are not that far off from the optimal one. Obviously the classic β approach outperforms p-norm by giving us optimal auxiliary signal for linear case. However, p-norm code is expected to run faster since we do not have to find optimal β. It indicates that if in some cases we have time constraint, or if it takes too long for β approach to find the answer, p-norm might be an alternative option of finding a good estimate of our auxiliary signal. For this specific case, the time difference was not significantly different (as shown in table 3.2) to support above stipulation.

Table 3.1: Cost and noise bound of auxiliary signal for linear Example 4.2.2 of [7]

Cost function				Noise bound			
lin.method	1-norm	2-norm	5-norm	lin.method	1-norm	2-norm	5-norm
3632	4065	3839	3762	1.00	1.12	1.06	1.04

Table 3.2: SOCS time required to calculate auxiliary signal for linear Example 4.2.2 of [7]

Time to find auxiliary signal [s]			
lin.method	1-norm	2-norm	5-norm
21.6	8.7	17.1	19.3

In the following sections we are going to use the above proposal to develop direct optimization formulation for systems of nonlinear differential equations ready to be coded into optimal control solver.

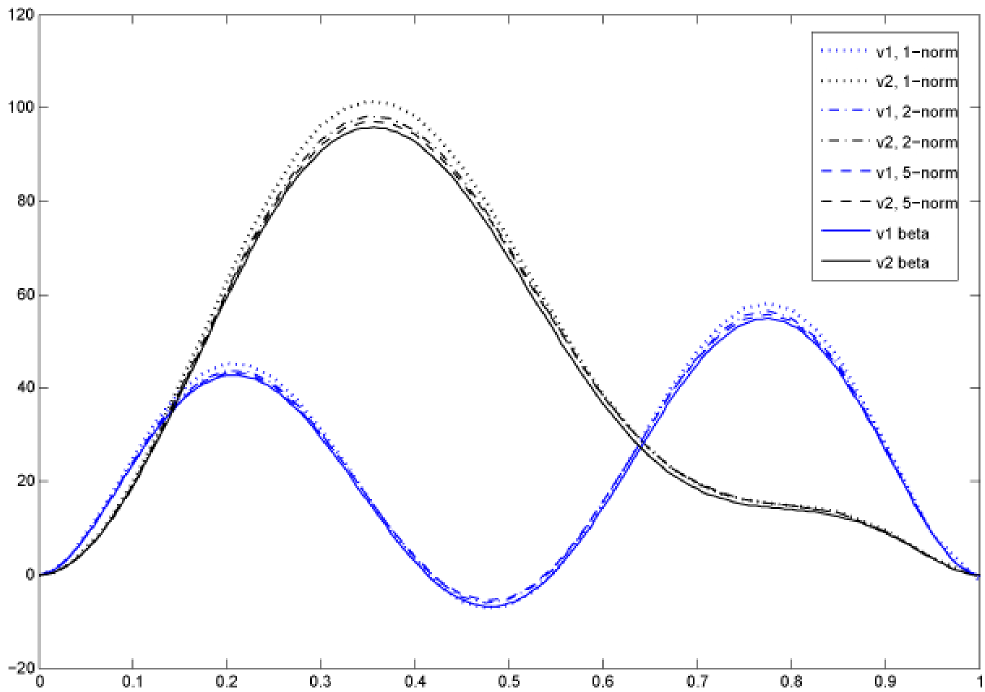

Figure 3.2: Aux signals for linear Example 4.2.2 of [7]

3.2 Definition of Problems with Additive Noise Only

3.2.1 Nonlinear Problems

We are observing the additive noise case ignoring additional inputs u in general problem setup (3.1) and we are assuming $\Gamma = I$.

Without loss of generality we can set noise bound $\rho = 1$ since we can adjust the weight on the noise ν_i within definition of f_i for a given problem. To make sure that we are getting a proper signal using the p-norm approach, our bound has to be $2\rho^p = 2$. We also know that for additive noise case, time parameter $s = T$.

Hence, the optimization problem constraints become

$$\begin{aligned}
\dot{x_0} &= f_0(x_0, v) + M_0\nu_0 && (3.12a)\\
\dot{x_1} &= f_1(x_1, v) + M_1\nu_1 && (3.12b)\\
\dot{\psi} &= f(\psi, v) && (3.12c)\\
0 &= g_0(x_0, v) + N_0\nu_0 - g_1(x_1, v) - N_1\nu_1 && (3.12d)\\
\inf_{x_i,\nu_i,y} & \quad S_0^p + S_1^p \geq 2 && (3.12e)
\end{aligned}$$

where

$$S_i = (x_i(0) - x_0)^T P_i(x_i(0) - x_0) + \int_0^T \nu_i^T \nu_i \, dt \quad (3.13)$$

We are assuming that noise matrices N_i have full row rank which is not restriction.

Under this assumption, we know that there exists representation $N_i = [\overline{N_i} \; 0]$ with $\overline{N_i}$ invertible, achievable by successive orthogonal changes of coordinates. Applying the same decomposition on matrices M_i and vectors ν_i, we have representations $M_i = [\overline{M_i} \; \widetilde{M_i}]$ and $\nu_i = [\overline{\nu_i}^T \; \widetilde{\nu_i}^T]^T$.

Now we can use (3.12d) to solve for $\overline{\nu_0}$

$$\overline{\nu_0} = \overline{N_0}^{-1} g_1(x_1, v) + \overline{N_0}^{-1}\overline{N_1}\,\overline{\nu_1} - \overline{N_0}^{-1} g_0(x_0, v) \quad (3.14)$$

and eliminate $\overline{\nu_0}$ from other equations to obtain the system of constraints

$$\begin{aligned}
\dot{x_0} &= f_0(x_0, v) + \\
&\quad + \overline{M_0}(\overline{N_0}^{-1} g_1(x_1, v) + \overline{N_0}^{-1}\overline{N_1}\,\overline{\nu_1} - \overline{N_0}^{-1} g_0(x_0, v)) + \widetilde{M_0}\widetilde{\nu_0} && (3.15a)\\
\dot{x_1} &= f_1(x_1, v) + \overline{M_1}\,\overline{\nu_1} + \widetilde{M_1}\widetilde{\nu_1} && (3.15b)\\
\dot{\psi} &= f(\psi, v) && (3.15c)\\
\inf_{x_i,\nu_i,y} & \quad S_0^p + S_1^p \geq 2 && (3.15d)\\
S_0 &= x_0(0)^T P_0 x_0(0) + \|\widetilde{\nu_0}\|_2^2 \\
&\quad + \int_0^T \left\| \overline{N_0}^{-1} g_1(x_1, v) + \overline{N_0}^{-1}\overline{N_1}\,\overline{\nu_1} - \overline{N_0}^{-1} g_0(x_0, v)) \right\|_2^2 dt && (3.15e)\\
S_1 &= x_1(0)^T P_1 x_1(0) + \int_0^T \|\overline{\nu_1}\|_2^2 + \|\widetilde{\nu_1}\|_2^2 \, dt && (3.15f)
\end{aligned}$$

The system of constraints (3.15) is in itself an optimization problem, denoted as the **inner optimization problem** from now on. To solve the inner optimization problem we define two new differential variables that are going to represent the uncertainty of the system:

$$\dot{z}_0 = \left\| \overline{N_0}^{-1} g_1(x_1, v) + \overline{N_0}^{-1} \overline{N_1} \overline{\nu_1} - \overline{N_0}^{-1} g_0(x_0, v) \right\|_2^2 + \|\widetilde{\nu_0}\|_2^2 \quad (3.16a)$$

$$\dot{z}_1 = \|\overline{\nu_1}\|_2^2 + \|\widetilde{\nu_1}\|_2^2 \quad (3.16b)$$

satisfying boundary conditions:

$$z_0(0) = x_0(0)^T P_0 x_0(0) \quad (3.17a)$$

$$z_1(0) = x_1(0)^T P_1 x_1(0) \quad (3.17b)$$

Rewritten inner minimization problem is to minimize $z_0(T)^p + z_1(T)^p$ with constraints is then

$$\dot{x}_0 = f_0(x_0, v) + \overline{M_0}(\overline{N_0}^{-1} g_1(x_1, v) + \overline{N_0}^{-1} \overline{N_1} \overline{\nu_1} - \overline{N_0}^{-1} g_0(x_0, v)) + \widetilde{M_0} \widetilde{\nu_0} \quad (3.18a)$$

$$\dot{x}_1 = f_1(x_1, v) + \overline{M_1} \overline{\nu_1} + \widetilde{M_1} \widetilde{\nu_1} \quad (3.18b)$$

$$\dot{z}_0 = \left\| \overline{N_0}^{-1} g_1(x_1, v) + \overline{N_0}^{-1} \overline{N_1} \overline{\nu_1} - \overline{N_0}^{-1} g_0(x_0, v) \right\|_2^2 + \|\widetilde{\nu_0}\|_2^2 \quad (3.18c)$$

$$\dot{z}_1 = \|\overline{\nu_1}\|_2^2 + \|\widetilde{\nu_1}\|_2^2 \quad (3.18d)$$

Following the process described in Section 2.1, we can derive the system of equations that represents the necessary conditions for the inner minimization problem. We are trying to minimize J where $J = \phi(x(0), x(T)) + \int_0^T L(x(t), u(t), t)\, dt$, with constraints

$$\dot{x} = f(x, u, t) \quad (3.19)$$

where $x = [x_0^T, x_1^T, z_0^T, z_1^T]^T$ and $u = [\widetilde{\nu_0}^T, \overline{\nu_1}^T, \widetilde{\nu_1}^T]^T$.

After introducing the Hamiltonian as $H(x, u, t) = L(x, u, t) + \lambda^T f(x, u, t)$, and using Leibnitz's rule and integration by parts we obtained the following increment of augmented performance index

$$dJ_a = (\phi_x - \lambda)^T dx|_{t=T} + (\phi_x + \lambda)^T dx|_{t=0} + \\ + \int_0^T ((H_x + \dot{\lambda})^T \delta x + H_u^T \delta u + (H_\lambda - \dot{x})^T \delta\lambda)\ dt \quad (3.20)$$

We noticed that starting and ending times are fixed. Theory states that to have minimum, $dJ_a = 0$ for all **independent increments**. Hence we get that necessary conditions are:

$$\dot{x} = f(x, u, t) \quad (3.21a)$$

$$\dot{\lambda} = -f_x^T \lambda - L_x \quad (3.21b)$$

$$0 = L_u + f_u^T \lambda \quad (3.21c)$$

$$0 = (\phi_x - \lambda)^T dx|_{t=T} \quad (3.21d)$$

$$0 = (\phi_x + \lambda)^T dx|_{t=0} \quad (3.21e)$$

For our specific type of problem we have

1. $L = 0$
2. $\phi = z_0(T)^p + z_1(T)^p$
3. Boundary conditions $z_0(0) = x_0(0)^T P_0 x_0(0)$ and $z_1(0) = x_1(0)^T P_1 x_1(0)$

Observe that because of the boundary conditions for $t = 0$, increments dx_0 and dz_0 are not independent, and dx_1 and dz_1 are not independent. That will create a boundary constraint by equation (3.21e). Including the above observations, our system of equations becomes:

$$
\begin{aligned}
\dot{x_0} &= f_0(x_0,v) + \overline{M_0}(\overline{N_0}^{-1} g_1(x_1,v) + \overline{N_0}^{-1}\overline{N_1}\overline{\nu_1} - \overline{N_0}^{-1} g_0(x_0,v)) \\
&\quad + \widetilde{M_0}\widetilde{\nu_0} && \text{(3.22a)} \\
\dot{x_1} &= f_1(x_1,v) + \overline{M_1}\overline{\nu_1} + \widetilde{M_1}\widetilde{\nu_1} && \text{(3.22b)} \\
\dot{z_0} &= \left\| \overline{N_0}^{-1} g_1(x_1,v) + \overline{N_0}^{-1}\overline{N_1}\overline{\nu_1} - \overline{N_0}^{-1} g_0(x_0,v) \right\|_2^2 + \|\widetilde{\nu_0}\|_2^2 && \text{(3.22c)} \\
\dot{z_1} &= \|\overline{\nu_1}\|_2^2 + \|\widetilde{\nu_1}\|_2^2 && \text{(3.22d)} \\
\dot{\lambda_0} &= (-(\frac{\partial f_0}{\partial x_0})^T + (\overline{M_0 N_0}^{-1} \frac{\partial g_0}{\partial x_0})^T)\lambda_0 \\
&\quad + 2(\overline{N_0}^{-1}\frac{\partial g_0}{\partial x_0})^T \cdot (\overline{N_0}^{-1} g_1(x_1,v) + \overline{N_0}^{-1}\overline{N_1}\overline{\nu_1} - \overline{N_0}^{-1} g_0(x_0,v))\lambda_2 && \text{(3.22e)} \\
\dot{\lambda_1} &= -(\overline{M_0 N_0}^{-1}\frac{\partial g_1}{\partial x_1})^T \lambda_0 - \frac{\partial f_1}{\partial x_1}^T \lambda_1 \\
&\quad - 2(\overline{N_0}^{-1}\frac{\partial g_1}{\partial x_1})^T \cdot (\overline{N_0}^{-1} g_1(x_1,v) + \overline{N_0}^{-1}\overline{N_1}\overline{\nu_1} - \overline{N_0}^{-1} g_0(x_0,v))\lambda_2 && \text{(3.22f)} \\
\dot{\lambda_2} &= 0 && \text{(3.22g)} \\
\dot{\lambda_3} &= 0 && \text{(3.22h)} \\
0 &= \widetilde{M_0}^T \lambda_0 + 2\widetilde{\nu_0}\lambda_2 && \text{(3.22i)} \\
0 &= (\overline{M_0 N_0}^{-1}\overline{N_1})^T \lambda_0 + \overline{M_1}^T \lambda_1 + 2(\overline{N_0}^{-1}\overline{N_1})^T \cdot (\overline{N_0}^{-1} g_1(x_1,v) \\
&\quad + \overline{N_0}^{-1}\overline{N_1}\overline{\nu_1} - \overline{N_0}^{-1} g_0(x_0,v))\lambda_2 + 2\overline{\nu_1}\lambda_3 && \text{(3.22j)} \\
0 &= \widetilde{M_1}^T \lambda_1 + 2\widetilde{\nu_1}\lambda_3 && \text{(3.22k)}
\end{aligned}
$$

With boundary conditions

$$x_0(0)^T P_0 x_0(0) = z_0(0) \tag{3.23a}$$

$$x_1(0)^T P_1 x_1(0) = z_1(0) \tag{3.23b}$$

$$p \cdot z_0(T)^{p-1} - \lambda_2(T) = 0 \tag{3.23c}$$

$$p \cdot z_1(T)^{p-1} - \lambda_3(T) = 0 \tag{3.23d}$$

$$\lambda_0(0) + 2P_0 x_0(0)\lambda_2(0) = 0 \tag{3.23e}$$

$$\lambda_1(0) + 2P_1 x_1(0)\lambda_3(0) = 0 \tag{3.23f}$$

$$\lambda_0(T) = 0 \tag{3.23g}$$

$$\lambda_1(T) = 0 \tag{3.23h}$$

The system of equations (3.22) and (3.23) solves the inner minimization problem. Combining those equations with our original optimization problem we obtain our final definition of the problem.

$$\text{minimize } \psi(T)^T W \psi(T) + \int_0^T |v|^2 + \psi^T U \psi \; dt$$

subject to

$$\begin{aligned}
\dot{x_0} &= f_0(x_0, v) + \overline{M_0}(\overline{N_0}^{-1} g_1(x_1, v) + \overline{N_0}^{-1}\overline{N_1}\overline{\nu_1} - \overline{N_0}^{-1} g_0(x_0, v)) \\
&\quad + \widetilde{M_0}\widetilde{\nu_0} && \text{(3.24a)} \\
\dot{x_1} &= f_1(x_1, v) + \overline{M_1}\overline{\nu_1} + \widetilde{M_1}\widetilde{\nu_1} && \text{(3.24b)} \\
\dot{z_0} &= \left\| \overline{N_0}^{-1} g_1(x_1, v) + \overline{N_0}^{-1}\overline{N_1}\overline{\nu_1} - \overline{N_0}^{-1} g_0(x_0, v) \right\|_2^2 + \|\widetilde{\nu_0}\|_2^2 && \text{(3.24c)} \\
\dot{z_1} &= \|\overline{\nu_1}\|_2^2 + \|\widetilde{\nu_1}\|_2^2 && \text{(3.24d)} \\
\dot{\lambda_0} &= (-(\frac{\partial f_0}{\partial x_0})^T + (\overline{M_0 N_0}^{-1} \frac{\partial g_0}{\partial x_0})^T)\lambda_0 \\
&\quad + 2(\overline{N_0}^{-1} \frac{\partial g_0}{\partial x_0})^T \cdot (\overline{N_0}^{-1} g_1(x_1, v) + \overline{N_0}^{-1}\overline{N_1}\overline{\nu_1} - \overline{N_0}^{-1} g_0(x_0, v))\lambda_2 && \text{(3.24e)} \\
\dot{\lambda_1} &= -(\overline{M_0 N_0}^{-1} \frac{\partial g_1}{\partial x_1})^T \lambda_0 - \frac{\partial f_1}{\partial x_1}^T \lambda_1 \\
&\quad - 2(\overline{N_0}^{-1} \frac{\partial g_1}{\partial x_1})^T \cdot (\overline{N_0}^{-1} g_1(x_1, v) + \overline{N_0}^{-1}\overline{N_1}\overline{\nu_1} - \overline{N_0}^{-1} g_0(x_0, v))\lambda_2 && \text{(3.24f)} \\
\dot{\lambda_2} &= 0 && \text{(3.24g)} \\
\dot{\lambda_3} &= 0 && \text{(3.24h)} \\
\dot{\psi} &= f_0(\psi, v) && \text{(3.24i)} \\
0 &= \widetilde{M_0}^T \lambda_0 + 2\widetilde{\nu_0}\lambda_2 && \text{(3.24j)} \\
0 &= (\overline{M_0 N_0}^{-1}\overline{N_1})^T \lambda_0 + \overline{M_1}^T \lambda_1 + 2(\overline{N_0}^{-1}\overline{N_1})^T \cdot (\overline{N_0}^{-1} g_1(x_1, v) \\
&\quad + \overline{N_0}^{-1}\overline{N_1}\overline{\nu_1} - \overline{N_0}^{-1} g_0(x_0, v))\lambda_2 + 2\overline{\nu_1}\lambda_3 && \text{(3.24k)} \\
0 &= \widetilde{M_1}^T \lambda_1 + 2\widetilde{\nu_1}\lambda_3 && \text{(3.24l)} \\
2 &\leq {z_0}^p(T) + {z_1}^p(T) && \text{(3.24m)}
\end{aligned}$$

with boundary conditions

$$x_0(0)^T P_0 x_0(0) = z_0(0) \tag{3.25a}$$

$$x_1(0)^T P_1 x_1(0) = z_1(0) \tag{3.25b}$$

$$\psi(0) = 0 \tag{3.25c}$$

$$p \cdot z_0(T)^{p-1} - \lambda_2(T) = 0 \tag{3.25d}$$

$$p \cdot z_1(T)^{p-1} - \lambda_3(T) = 0 \tag{3.25e}$$

$$\lambda_0(0) + 2P_0 x_0(0)\lambda_2(0) = 0 \tag{3.25f}$$

$$\lambda_1(0) + 2P_1 x_1(0)\lambda_3(0) = 0 \tag{3.25g}$$

$$\lambda_0(T) = 0 \tag{3.25h}$$

$$\lambda_1(T) = 0 \tag{3.25i}$$

The system (3.24) and (3.25) of differential algebraic equations with boundary constraints is ready to be applied on the specific problem and be fed into some optimal control solver like SOCS.

3.2.2 Linear Problems

The theory for linear problems and applications was already developed and presented in Chapter 2. But even then, the p-norm approach can be applicable as an alternative approach. Trying the p-norm approach would make sense if we are willing to sacrifice optimality of the proper signal v to save some time or if for any reason the standard approach does not provide result.

For equation derivation purposes, we just have to notice that this is a special case of Section 3.2.1 where functions f_i and g_i have linear form. We can use all assumptions we used in Section 3.2.1. In addition, we assume that input signal v is not part of output equations.

Therefore, our problem is to minimize

$$\psi(T)^T W \psi(T) + \int_0^T |v|^2 + \psi^T U \psi \; dt$$

subject to

$$\dot{x_0} = A_0 x_0 + B_0 v + M_0 \nu_0 \quad (3.26a)$$

$$\dot{x_1} = A_1 x_1 + B_1 v + M_1 \nu_1 \quad (3.26b)$$

$$\dot{\psi} = A_0 \psi + B_0 v \quad (3.26c)$$

$$0 = C_0 x_0 + N_0 \nu_0 - C_1 x_1 - N_1 \nu_1 \quad (3.26d)$$

$$\inf_{x_i, \nu_i, y} \quad S_0^p + S_1^p \geq 2 \quad (3.26e)$$

where

$$S_i = (x_i(0) - x_0)^T P_i (x_i(0) - x_0) + \int_0^T \nu_i^T \nu_i \; dt \quad (3.27)$$

By replacing f_0, f_1, g_0 and g_1 from system of equations for case I with corresponding matrices in (3.24) we obtain the optimal problem definition for this case.

$$\text{minimize } \psi(T)^T W \psi(T) + \int_0^T |v|^2 + \psi^T U \psi \; dt$$

subject to

$$\dot{x_0} = A_0 x_0 + B_0 v + \overline{M_0}(\overline{N_0}^{-1} C_1 x_1 + \overline{N_0}^{-1}\overline{N_1}\overline{\nu_1} - \overline{N_0}^{-1} C_0 x_0) + \widetilde{M_0}\widetilde{\nu_0} \tag{3.28a}$$

$$\dot{x_1} = A_1 x_1 + B_0 v + \overline{M_1}\overline{\nu_1} + \widetilde{M_1}\widetilde{\nu_1} \tag{3.28b}$$

$$\dot{z_0} = \left\| \overline{N_0}^{-1} C_1 x_1 + \overline{N_0}^{-1}\overline{N_1}\overline{\nu_1} - \overline{N_0}^{-1} C_0 x_0 \right\|_2^2 + \|\widetilde{\nu_0}\|_2^2 \tag{3.28c}$$

$$\dot{z_1} = \|\overline{\nu_1}\|_2^2 + \|\widetilde{\nu_1}\|_2^2 \tag{3.28d}$$

$$\begin{aligned}\dot{\lambda_0} = {} & (-A_0^T + (\overline{M_0 N_0}^{-1} C_0)^T)\lambda_0 \\ & +2(\overline{N_0}^{-1} C_0)^T \cdot (\overline{N_0}^{-1} C_1 x_1 + \overline{N_0}^{-1}\overline{N_1}\overline{\nu_1} - \overline{N_0}^{-1} C_0 x_0)\lambda_2\end{aligned} \tag{3.28e}$$

$$\begin{aligned}\dot{\lambda_1} = {} & -(\overline{M_0 N_0}^{-1} C_1)^T \lambda_0 - A_1^T \lambda_1 \\ & -2(\overline{N_0}^{-1} C_1)^T \cdot (\overline{N_0}^{-1} C_1 x_1 + \overline{N_0}^{-1}\overline{N_1}\overline{\nu_1} - \overline{N_0}^{-1} C_0 x_0)\lambda_2\end{aligned} \tag{3.28f}$$

$$\dot{\lambda_2} = 0 \tag{3.28g}$$

$$\dot{\lambda_3} = 0 \tag{3.28h}$$

$$\dot{\psi} = A_0 \psi + B_0 v \tag{3.28i}$$

$$0 = \widetilde{M_0}^T \lambda_0 + 2\widetilde{\nu_0}\lambda_2 \tag{3.28j}$$

$$\begin{aligned}0 = {} & (\overline{M_0 N_0}^{-1}\overline{N_1})^T \lambda_0 + \overline{M_1}^T \lambda_1 \\ & +2(\overline{N_0}^{-1}\overline{N_1})^T (\overline{N_0}^{-1} C_1 x_1 + \overline{N_0}^{-1}\overline{N_1}\overline{\nu_1} - \overline{N_0}^{-1} C_0 x_0)\lambda_2 + 2\overline{\nu_1}\lambda_3\end{aligned} \tag{3.28k}$$

$$0 = \widetilde{M_1}^T \lambda_1 + 2\widetilde{\nu_1}\lambda_3 \tag{3.28l}$$

$$2 \leq {z_0}^p(T) + {z_1}^p(T) \tag{3.28m}$$

with boundary conditions

$$z_0(0) - x_0(0)^T P_0 x_0(0) = 0 \tag{3.29a}$$
$$z_1(0) - x_1(0)^T P_1 x_1(0) = 0 \tag{3.29b}$$
$$\psi(0) = 0 \tag{3.29c}$$
$$p \cdot z_0(T)^{p-1} - \lambda_2(T) = 0 \tag{3.29d}$$
$$p \cdot z_1(T)^{p-1} - \lambda_3(T) = 0 \tag{3.29e}$$
$$\lambda_0(0) + 2P_0 x_0(0)\lambda_2(0) = 0 \tag{3.29f}$$
$$\lambda_1(0) + 2P_1 x_1(0)\lambda_3(0) = 0 \tag{3.29g}$$
$$\lambda_0(T) = 0 \tag{3.29h}$$
$$\lambda_1(T) = 0 \tag{3.29i}$$

The system (3.28) of differential algebraic equations with boundary constraints (3.29) is ready to be applied on a specific problem and be fed into some optimal control solver like SOCS.

3.3 Linear Problems with Model and Additive Uncertainty

In Section 2.3.2 we presented one way to describe a system with model uncertainty.

$$\dot{x} = Ax + Bv + K\mu + M\nu \tag{3.30a}$$
$$y = Cx + Dv + L\mu + N\nu \tag{3.30b}$$
$$\xi = Gx + Hv \tag{3.30c}$$
$$S = (x(0) - x_0)^T P_0^{-1}(x(0) - x_0) + \int_0^s |\nu|^2 + |\mu|^2 - |\xi|^2 \; dt < 1 \tag{3.30d}$$

$\forall s \in [0, T]$ with $\mu = \Delta\xi$ and function Δ having the largest singular value smaller than 1. Notice the change in the names of some matrices. Our original notation from

(2.32) followed literature convention [7] but is not convenient across problem types. The new notation is done to keep consistency with additive noise models and to make new MATLAB and MAPLE routines compatible. In this thesis we will assume that noise matrices N have a full row rank and that there is no model uncertainty in v, that is $H = 0$. Without loss of generality we take that the noise bound is 1.

We are going now to derive formulas for the direct optimization method that uses parameter β just like it was done for additive uncertainty in Section 4.2.2 of [7]. Currently, the only codes that solve some problems with model uncertainty exist use a Riccati equations approach [7]. However, being able to define this problem as a big optimal control problem will allow us to solve other new types of problems, such as

1. Different cost functions
2. Additional constraints on the auxiliary signal v
3. Additional constraints on the states
4. Multiple fault models

The results using Riccati equations approach will be used to test our code.

Let's assume that we have 2 models, represented as above by using different indicies. That way we have our system of differential-algebraic constraints. Our goal is to minimize

$$\delta^2(v) = \psi(T)^T W \psi(T) + \int_0^T |v|^2 + \psi^T U \psi \; dt \tag{3.31}$$

subject to having at least one noise bound broken given the outputs y_i are the same.

Let

$$\phi_\beta(v) = \inf_{x_i,\varphi_i,y,s_i} \beta S_0(x_0(0), \varphi_1, s_0) + (1-\beta) S_1(x_1(0), \varphi_1, s_1) \geq 1 \tag{3.32}$$

where

$$S_i = (x_i(0) - x_{i0})^T P_i (x_i(0) - x_{i0}) + \int_0^{s_i} \varphi_i^T \Gamma \varphi_i \; dt \tag{3.33}$$

with $\varphi_i = \begin{bmatrix} \nu_i \\ \mu_i \\ \xi_i \end{bmatrix}$ and $\Gamma = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & -I \end{bmatrix}$.

Differential-algebraic path constraints are:

$$\dot{x} = Ax + Bv + M\nu \quad (3.34a)$$

$$0 = Cx + N\nu \quad (3.34b)$$

$$0 = Gx - \xi \quad (3.34c)$$

where $A = \begin{bmatrix} A_0 & 0 \\ 0 & A_1 \end{bmatrix}$,$B = \begin{bmatrix} B_0 \\ B_1 \end{bmatrix}$, $M = \begin{bmatrix} K_0 & M_0 & 0 & 0 \\ 0 & 0 & K_1 & M_1 \end{bmatrix}$,

$C = \begin{bmatrix} C_0 & -C_1 \end{bmatrix}$,$N = \begin{bmatrix} L_0 & N_0 & -L_1 & -N_1 \end{bmatrix}$, $G = \begin{bmatrix} G_0 & 0 \\ 0 & G_1 \end{bmatrix}$,

$\nu = \begin{bmatrix} \mu_0 \\ \nu_0 \\ \mu_1 \\ \nu_1 \end{bmatrix}$ and $\begin{bmatrix} x_0 \\ x_1 \end{bmatrix}$.

Since we are using direct transcription method, it is beneficial to reduce the problem by eliminating some algebraic constraints. Since N_i parts of matrix N have a full row rank, we can perform a constant orthogonal change of additive noise components ν_i in ν resulting in having $N_i = [\overline{N_i}, 0]$ with $\overline{N_i}$ being invertible. Using the same decomposition, we will have $M_i = [\overline{M_i}, \tilde{M_i}]$ and $\nu_i = [\overline{\nu_i}^T, \tilde{\nu_i}^T]^T$. Now we can use algebraic equation (3.34b) to solve for $\overline{\nu_0}$ and eliminate those variables from the other equations (Note, we could choose to eliminate $\overline{\nu_1}$ instead). Solving for $\overline{\nu_0}$ will involve redefining noise vector, and having all matrices involved multiplied by $\overline{N_0}^{-1}$. To keep notation as simple as possible we will still call $\overline{N_0}^{-1}C$ just C, and the same with remaining parts of the matrix N.

After substituting, our system becomes

$$\dot{x} = Ax + Bv + M\nu \quad (3.35)$$

with $A = \begin{bmatrix} A_0 - \overline{M_0}C_0 & \overline{M_0}C_1 \\ 0 & A_1 \end{bmatrix}$, $B = \begin{bmatrix} B_0 \\ B_1 \end{bmatrix}$ and

$M = \begin{bmatrix} K_0 - \overline{M_0}L_0 & \tilde{M}_0 & \overline{M_0}L_1 & \overline{M_0 N_1} & 0 \\ 0 & 0 & K_1 & \overline{M_1} & \tilde{M}_1 \end{bmatrix}$ and a new noise vector

$\nu = [\mu_0^T, \tilde{\nu_0}^T, \mu_1^T, \overline{\nu_1}^T, \tilde{\nu_1}^T]^T$.

Equation (3.34c) is used to solve for ξ. So the new cost of the $\phi_\beta(v)$ is of the form

$$x(0)^T P_\beta x(0) + \frac{1}{2}\int_0^s x^T Q x + x^T H\nu + \nu^T R\nu \ dt \tag{3.36}$$

with $Q = 2\begin{bmatrix} \beta C_0^T C_0 - \beta G_0^T G_0 & -\beta C_0^T C_1 \\ -\beta C_1^T C_0 & \beta C_1^T C_1 - (1-\beta)G_1^T G_1 \end{bmatrix}$,

$H = 4\beta \begin{bmatrix} C_0^T L_0 & 0 & -C_0^T L_1 & -C_0^T\overline{N_1} & 0 \\ -C_1^T L_0 & 0 & C_1^T L_1 & C_1^T \overline{N_1} & 0 \end{bmatrix}$,

$R = 2\begin{bmatrix} \beta I + \beta L_0^T L_0 & 0 & -\beta L_0^T L_1 & -\beta L_0^T \overline{N_1} & 0 \\ 0 & \beta I & 0 & 0 & 0 \\ -\beta L_1^T L_0 & 0 & \beta L_1^T L_1 + (1-\beta)I & \beta L_1^T \overline{N_1} & 0 \\ -\beta \overline{N_1}^T L_0 & 0 & \beta \overline{N_1}^T L_1 & \beta \overline{N_1}^T \overline{N_1} + (1-\beta)I & 0 \\ 0 & 0 & 0 & 0 & (1-\beta)I \end{bmatrix}$

and $P_\beta = diag\{\beta P_0, (1-\beta)P_1\}$.

Finally, after introducing new variable z to account for noise, allow more generalized cost function by letting the influence of auxiliary signal on the system be part of it through variable ψ, and applying Lagrange multipliers λ with Hamiltonian

$$\overline{H(t)} = \frac{1}{2}(x^T Q x + x^T H\nu + \nu^T R\nu) + \lambda^T(Ax + Bv + M) \tag{3.37}$$

we obtain the following description of the problem:

$$\text{minimize } \delta^2(v) = \psi(T)^T W \psi(T) + \int_0^T |v|^2 + \psi^T U \psi \ dt$$

subject to path constraints

$$\dot{x} = Ax + Bv + M\nu \tag{3.38a}$$

$$\dot{\psi} = A_0\psi + B_0 v \tag{3.38b}$$

$$\dot{\lambda} = -Qx - \frac{1}{2}H\nu - A^T\lambda \tag{3.38c}$$

$$\dot{z} = \frac{1}{2}(x^T Qx + x^T H\nu + \nu^T R\nu) \tag{3.38d}$$

$$0 = R\nu + \frac{1}{2}H^T x + M^T\lambda \tag{3.38e}$$

and boundary conditions and other constraints

$$\psi(0) = 0 \tag{3.39a}$$

$$\lambda(0) + 2P_\beta x(0) = 0 \tag{3.39b}$$

$$z(0) = x(0)^T P_\beta x(0) \tag{3.39c}$$

$$\lambda(s) = 0 \tag{3.39d}$$

$$0 < \beta < 1 \tag{3.39e}$$

$$0 \leq s \leq T \tag{3.39f}$$

$$z(s) \geq 1 \tag{3.39g}$$

After a brief discussion about the drivers, we will present some test examples where we will compare results obtained using direct optimization method and the method using Riccati equations.

3.3.1 Software Implementation

We developed drivers in MATLAB and MAPLE that provide system of differential-algebraic equations ready to be fed into the SOCS driver. These drivers were inspired by the previous drivers written for additive noise case [25] and are using β approach. We also needed to modify the SOCS drivers. The main issue is dealing with the issue that our integrand in the cost function is defined over the whole testing interval $[0, T]$

$$\text{minimize } \delta^2(v) = \psi(T)^T W \psi(T) + \int_0^T |v|^2 + \psi^T U \psi \, dt$$

while our condition for noise bound depends on the parameter $s_i \in [0, T]$

$$S_i = (x_i(0) - x_{i0})^T P_i^{-1} (x_i(0) - x_{i0}) + \int_0^{s_i} \varphi_i^T \Gamma \varphi_i \, dt \tag{3.40}$$

For additive noise we did not have this issue since matrix Γ was positive definite, so we were able to set $s_i = T$. For model uncertainty problems that is not the case since Γ (3.33) is not positive definite anymore.

Still, modifications of the SOCS driver for cases where $U = W = 0$ are not too different from the additive uncertainty cases. We can set the end of the interval to be variable the $T = s$, where s is one of the values s_i for which one of the models breaks the noise bound. We can do that since for $t > s$, we already know which model we are dealing with, so $v = 0$ is the optimal solution for $t > s$. Though modification of the driver is simple, due to the nonlinearities in s, we will encounter local minimums depending on the initial guess for s.

When either of matrices U or W is not 0, the problem is more difficult since cost function now depends on ψ. Hence, even for $t > s$, v signal can help reduce the size of ψ, so $v = 0$ after $t = s$ is not necessarily an optimal solution anymore. Therefore we have to account in our code for both T and s. We solved that problem in SOCS by splitting the problem in 2 phases. The first phase is very similar to the case when $U = W = 0$, with final time of the phase 1 being variable s. We use the system of equations derived in this section as constraints for this phase with one modification. Since we need the value of the cost function on interval $0 \leq t \leq s$ to be passed to phase 2, we define the new variable

$$\dot{\omega} = |v|^2 + \psi^T U \psi \tag{3.41}$$

with an initial condition $\omega(t_0) = 0$.

Phase 2 is defined on the time interval $s \le t \le T$. Since we know that the noise bound is already broken, we only care to account on the influence of v on ψ and how is the cost represented by ω affected. So the system of equations for phase 2 is much simpler:

$$\begin{aligned} \dot{\psi} &= A_0\psi + B_0 v && (3.42a) \\ \dot{\omega} &= |v|^2 + \psi^T U \psi && (3.42b) \end{aligned}$$

We have boundary conditions that require that variables ψ, ω and t are continuous from phase 1 to phase 2. Our overall goal is

$$\text{minimize } \psi(T)^T W \psi(T) + \omega(T)$$

3.3.2 Test Examples

This is the Example 3.3.1 from [7] which has 2 cases. We will compare results obtained by solving this problem represented as the optimization problem using SOCS, and results obtained using Riccati equations implemented in Scilab. Scilab is an open source platform for numerical computation developed by INRIA. More details about Scilab and this approach to the problem can be found in [8], [7] or at Scilab website www.scilab.org.

The First Case of Example 3.3.1

The system of differential-algebraic constraints with model uncertainty is defined as

$$\begin{aligned} \dot{x_0} &= \begin{bmatrix} 0.4\delta_1 & 1 \\ -1 + 0.4\delta_2 & 0 \end{bmatrix} x_0 + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} v + 10^{-4} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \nu_0 && (3.43a) \\ \dot{x_1} &= \begin{bmatrix} 0.4\delta_3 & 3 \\ -3 + 0.4\delta_4 & 0 \end{bmatrix} x_1 + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} v + 10^{-4} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \nu_1 && (3.43b) \\ 0 &= \begin{bmatrix} 1 & 2 \end{bmatrix} x_0 - \begin{bmatrix} 1 & 2 \end{bmatrix} x_1 + \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \nu_0 - \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \nu_1 && (3.43c) \end{aligned}$$

which translates in the following form using the procedure on pg.31 of [7]

$$\dot{x_0} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} x_0 + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} v + \begin{bmatrix} 1 & 0 & 10^{-4} & 0 & 0 \\ 0 & 1 & 0 & 10^{-4} & 0 \end{bmatrix} \nu_0 \quad (3.44a)$$

$$\dot{x_1} = \begin{bmatrix} 0 & 3 \\ -3 & 0 \end{bmatrix} x_1 + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} v + \begin{bmatrix} 1 & 0 & 10^{-4} & 0 & 0 \\ 0 & 1 & 0 & 10^{-4} & 0 \end{bmatrix} \nu_1 \quad (3.44b)$$

$$z_0 = \begin{bmatrix} 0.4 & 0 \end{bmatrix} x_0 \quad (3.44c)$$

$$z_1 = \begin{bmatrix} 0.4 & 0 \end{bmatrix} x_1 \quad (3.44d)$$

$$0 = \begin{bmatrix} 1 & 2 \end{bmatrix} x_0 - \begin{bmatrix} 1 & 2 \end{bmatrix} x_1 + \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \end{bmatrix} \nu_0 - \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \end{bmatrix} \nu_1 \quad (3.44e)$$

Notice the increase of dimensions of noise vectors ν_0 and ν_1 and their corresponding matrices.

The uncertainty for both models is

$$S_i(v, s) = |x_i(0)|^2 + \int_0^s |\nu_i(t)|^2 - |z_i(t)|^2 \; dt < 1 \quad (3.45)$$

where $0 \leq s \leq T$ and the cost function is

$$\delta^2(v) = \psi(T)^T W \psi(T) + \int_0^T |v|^2 + \psi^T U \psi \; dt \quad (3.46)$$

The above problem is then prepared for SOCS as described in Section 3.3.1.

First we will assume the simpler problem where $U = W = 0$ and $T = 10$. Calculated signals v are presented in figure 3.3. We can observe that the signals are very similar in shape. Both methods found that $s = 10$ gives the optimal proper signal v. The calculated value of parameter β and the cost functions are given in the table 3.3.

Table 3.3: Parameter comparison of SOCS and Scilab runs for $U = W = 0$

Parameter	β	Cost	Time(s)
SOCS	0.80	1.357	148.77
Scilab	.75	1.433	unavailable

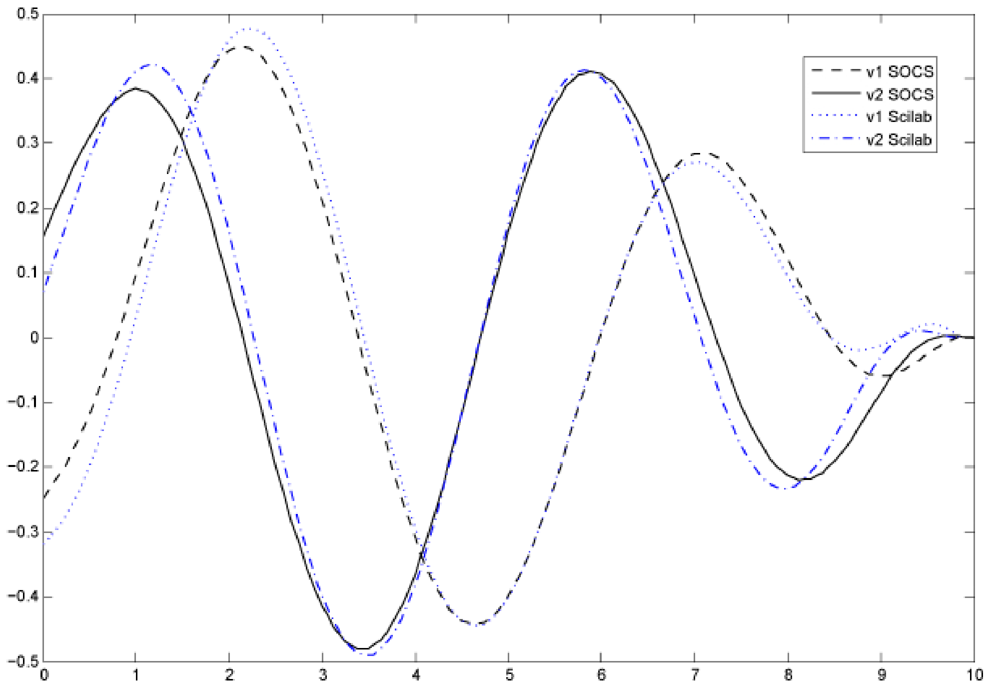

Figure 3.3: Comparison of signals v obtained by SOCS and Scilab for $U = W = 0$

Notice that cost is slightly improved by using SOCS approach. This might be due to the Scilab routine that uses β optimization on a fixed grid, while in SOCS β is a continuous parameter. Hence we might expect SOCS to be slightly more accurate in some cases. The time of 148.77s from the table is obtained when the constraint on variable s was $0.1 \leq s \leq 10$. We did runs for different lower bounds of s, and we obtained the same solution but in shorter time. This pointed out again the importance of good initial guess. The more drastic difference obtained by choosing different initial guess is shown in the case 2 of the problem 3.3.1.

Now we will focus on the computationally more difficult case where $U = W = I$ and $T = 10$. The graphs of v are in figure 3.4 and parameters are in the table 3.4. The graphs are of a similar shape again, with the small improvement of cost function when using SOCS. The time represented was for run where s had only upper bound of $s \leq 9.99$. The reason why the upper bound is not 10, is due to the SOCS requirement

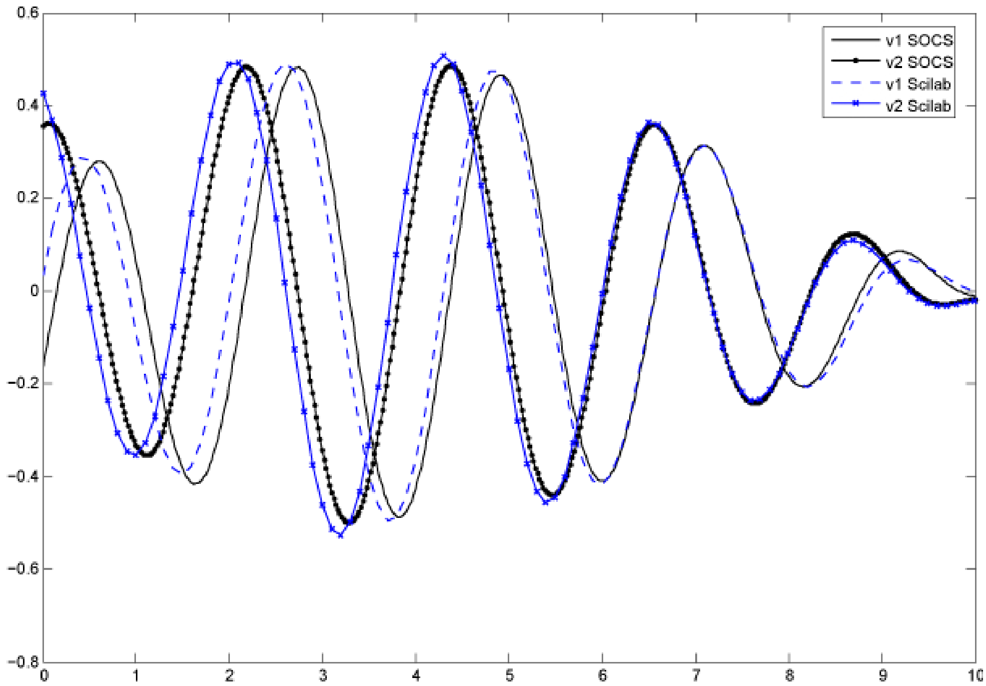

Figure 3.4: Comparison of signals v obtained by SOCS and Scilab for $U = W = I$

Table 3.4: Parameter comparison of SOCS and Scilab runs for $U = W = I$

Parameter	β	Cost	Time(s)
SOCS	0.185	1.838	116.5
Scilab	.247	1.990	unavailable

that phase length exists for each phase.

Overall, the comparison of the existing results obtained through Scilab with the results we obtained by direct optimization approach using SOCS confirmed that our code works. SOCS even slightly outperformed Scilab. However, when using SOCS we have an issue with choosing initial guess and local minimums showing up due to the nonlinearity in s. That is shown in the next case

The Second Case of Example 3.3.1

The second case is very similar to the first case except that problem is simplified by having $\delta_2 = \delta_4 = 0$. The formulation of the problem is the following:

$$\dot{x_0} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} x_0 + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} v + \begin{bmatrix} 1 & 0 & 10^{-4} & 0 & 0 \\ 0 & 0 & 0 & 10^{-4} & 0 \end{bmatrix} \nu_0 \tag{3.47a}$$

$$\dot{x_1} = \begin{bmatrix} 0 & 3 \\ -3 & 0 \end{bmatrix} x_1 + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} v + \begin{bmatrix} 1 & 0 & 10^{-4} & 0 & 0 \\ 0 & 0 & 0 & 10^{-4} & 0 \end{bmatrix} \nu_1 \tag{3.47b}$$

$$z_0 = \begin{bmatrix} 0.4 & 0 \end{bmatrix} x_0 \tag{3.47c}$$

$$z_1 = \begin{bmatrix} 0.4 & 0 \end{bmatrix} x_1 \tag{3.47d}$$

$$\begin{aligned} 0 = {} & \begin{bmatrix} 1 & 2 \end{bmatrix} x_0 - \begin{bmatrix} 1 & 2 \end{bmatrix} x_1 \\ & + \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \end{bmatrix} \nu_0 - \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \end{bmatrix} \nu_1 \end{aligned} \tag{3.47e}$$

As in the first case, the uncertainty for both models is

$$S_i(v, s) = |x_i(0)|^2 + \int_0^s |\nu_i(t)|^2 - |z_i(t)|^2 \ dt < 1 \tag{3.48}$$

where $0 \leq s \leq T$ and the cost function is

$$\delta^2(v) = \psi(T)^T W \psi(T) + \int_0^T |v|^2 + \psi^T U \psi \ dt \tag{3.49}$$

We will consider only when $U = W = 0$ and $T = 10$. For this case, depending on the lower bound constraint L_b on s we obtained different values for s and hence

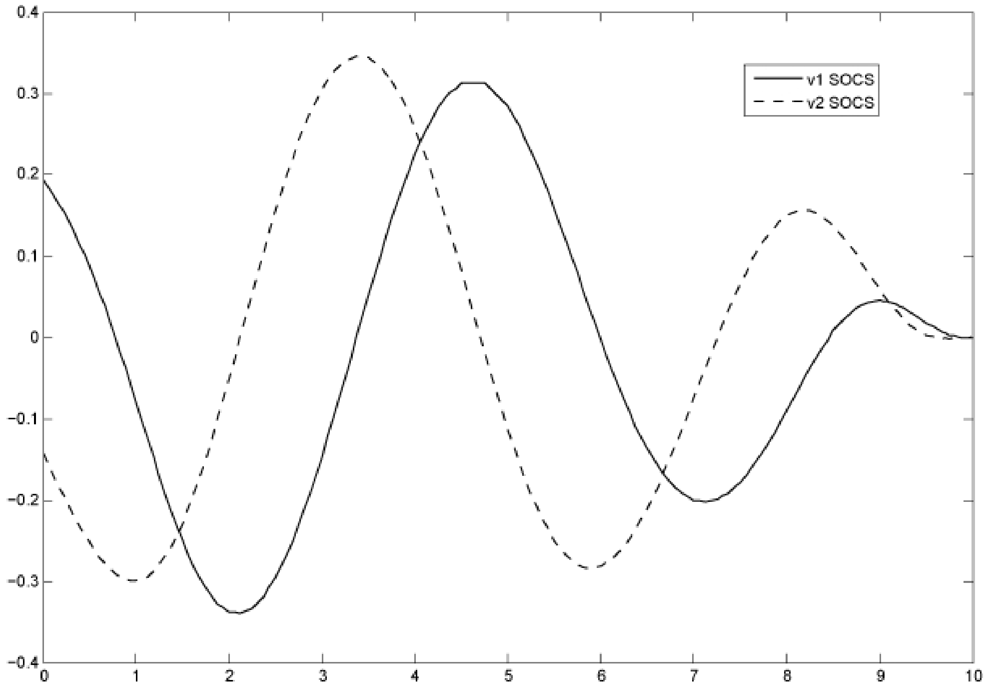

Figure 3.5: The best signals v found

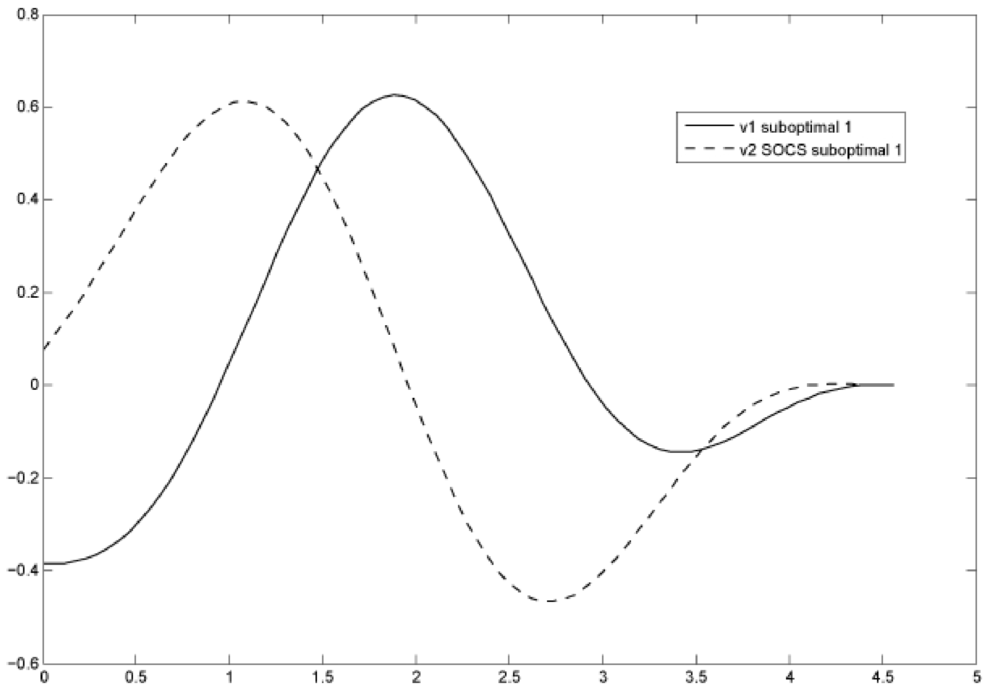

Figure 3.6: The suboptimal signal 1

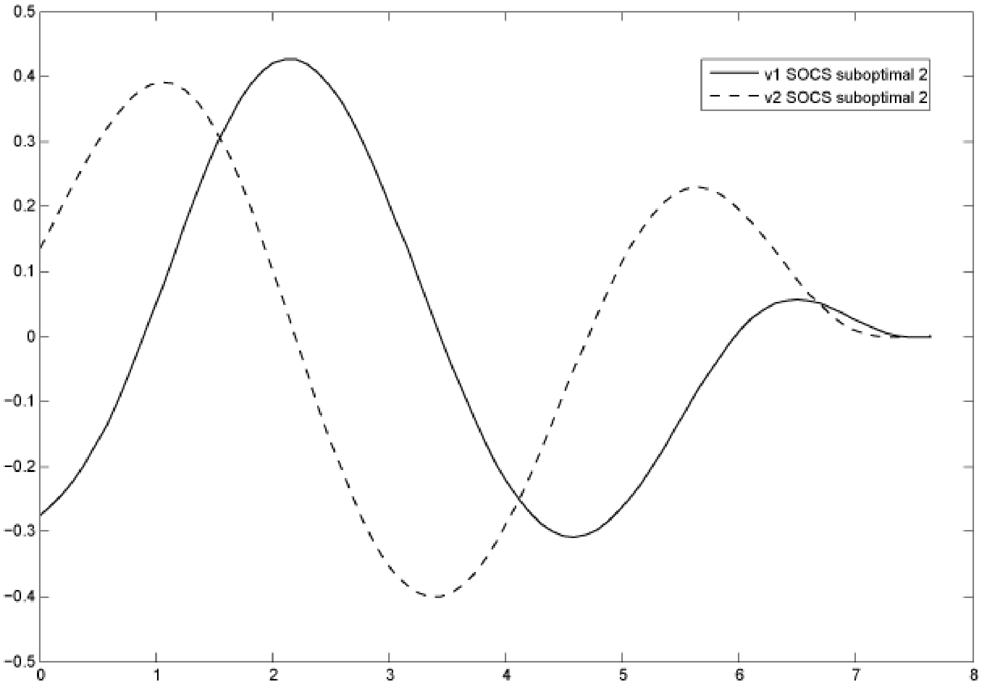

Figure 3.7: Suboptimal signal 2

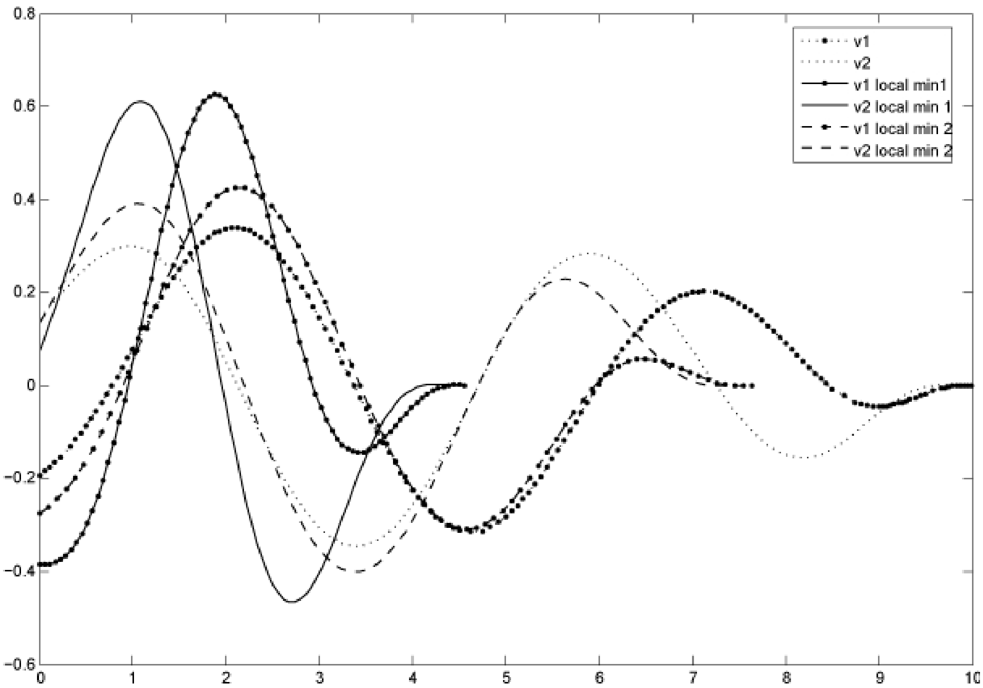

Figure 3.8: Comparison of the signals v

Table 3.5: Parameter comparison of different solutions for v

Parameter	β	Cost	Time(s)	L_b	s
The best v	0.773	0.725	144.8	7	10
Suboptimal 1	.631	0.998	32.5	0.1	4.56
Suboptimal 2	.750	.809	32.8	3	7.65

the different signals v as shown on figures 3.5, 3.6, 3.7 and 3.8. The performance parameters of these 3 calculated signals are in the table 3.5.

Initial guess of s was L_b. We can see that SOCS got stuck twice in a local minimum as a consequence of the different initial guess. It is interesting to note that while $L_b = 5, 7, 8$ or 9 gave us the best answer, $L_b = 6$ gave us the suboptimal 2 result. This points out that dependence on the initial guess is not trivial. One of the reasons why, is the way the initial guess for all other variables is defined in this case. Namely, they are just a linear function between $t = 0$ and $t = L_b$. The research on this effect will be a topic for future research.

3.4 Noise Bound Setup

Since using the p-norm approach produces a suboptimal auxiliary signal v, we needed some way to evaluate its performance. One way is to calculate for what noise bound the calculated v is proper.

Finding that bound B_d using SOCS is defined as the following parameter optimization problem:

$$\text{minimize parameter } B_d$$

subject to

$$\dot{x_0} = f_0(x_0, \nu_{0x}, v) \tag{3.50a}$$

$$\dot{x_1} = f_1(x_1, \nu_{1x}, v) \tag{3.50b}$$

$$\dot{z_0} = \nu_{0x}^T \nu_{0x} + \nu_{0y}^T \nu_{0y} \tag{3.50c}$$

$$\dot{z_1} = \nu_{1x}^T \nu_{1x} + \nu_{1y}^T \nu_{1y} \tag{3.50d}$$

$$0 = y_0 - y_1 = g_0(x_0, \nu_{0y}, v) - g_1(x_1, \nu_{1y}, v) \tag{3.50e}$$

with boundary conditions

$$z_0(0) = 0 \tag{3.51a}$$

$$z_1(0) = 0 \tag{3.51b}$$

$$B_d - x_0(0)^T W x_0(0) - z_0(T) \geq 0 \tag{3.51c}$$

$$B_d - x_1(0)^T W x_1(0) - z_1(T) \geq 0 \tag{3.51d}$$

In the above equations W is a matrix representing the weight on the initial condition. Variables $z_i, i = 0, 1$ are noise measures. Since in this thesis we were assuming that the noise bound is 1, our expectation is that calculated parameter B_d will be greater than 1, but hopefully not by much.

The precalculated auxiliary signal v was given to SOCS as a subroutine that evaluated v as follows:

1. Read current time t_c from SOCS
2. Read text file containing column representing t and column representing v to find between which 2 rows is t_c
3. Use the data from those 2 rows to interpolate value for v
4. Return this value to SOCS

A few final comments:

1. Common ODE tolerances used in SOCS were 10^{-4} and 10^{-5}. SOCS stops the grid refinement when it estimates that ODE tolerance has been met in the dynamics and the cost is to this tolerance also.

2. In most of the examples we considered here, finding noise bound did not take long (less then 2-3 minutes). However, we encountered examples where it took time in the order of days to finish this search. In Chapter 4 these occurrences are documented

3. For nonlinear cases, we have to be careful due to the existence of possible local minimums and maximums

4. Current implemented interpolation is just linear approximation. We did not have need to change that. In the future research this implementation can be changed. For instance, the newer version of SOCS provides its own interpolation subroutine

Chapter 4

Examples and Analysis

In this chapter, a few selected examples are going to be presented. In the first three examples we are focusing on nonlinear systems and corresponding linearized systems with additive uncertainty only. We will present advantages and disadvantages of the p-norm approach (Section 3.1.1) on the nonlinear problem, in comparison to the β approach on a linearized version.

The following examples will analyze linear problem with additive and model uncertainty. We will use various modifications of this problem to show the added capabilities that can be achieved by solving active fault detection as one big optimization problem. Such modifications include handling both, hard and soft constraints on the proper auxiliary signal v that the approach using Riccati equations cannot solve.

4.1 Example 1

This is a theoretical example that we thought will be useful for future research. The nonlinearity is quadratic in the state. Beside problems derived from physics formulas, where second power often appears, this type of nonlinearity can have more significant impact for future research. For practical purposes, we usually approximate nonlinearities by using only up to linear term in Taylor series in the process of linearization. It is obvious that improvement of the approximation, at the expense of complexity, could be achieved by using quadratic approximation where we will use quadratic term of the Taylor series as well. That would lead to the nonlinear problems of the same type as this example.

The formulation of our example is:

$$\text{minimize } \delta^2(v) = \psi(T)^T W \psi(T) + \int_0^T |v|^2 + \psi^T U \psi \; dt$$

for a proper v (as defined in Section 2.3.1), given the nonlinear problem

$$\dot{x_0} = -x_0^2 - x_0 + v + n_0 \widetilde{\nu_0} \tag{4.1a}$$

$$\dot{x_1} = -x_1^2 - 2x_1 + v + n_1 \widetilde{\nu_1} \tag{4.1b}$$

$$0 = x_1 - x_0 + \overline{\nu_1} - \overline{\nu_0} \tag{4.1c}$$

and its linearized version (linearized about $(x_0, x_1) = (0, 0)$)

$$\dot{x_0} = -x_0 + v + n_0 \widetilde{\nu_0} \tag{4.2a}$$

$$\dot{x_1} = -2x_1 + v + n_1 \widetilde{\nu_1} \tag{4.2b}$$

$$0 = x_1 - x_0 + \overline{\nu_1} - \overline{\nu_0} \tag{4.2c}$$

The noise perturbation is defined as in (3.13)

$$S_i = x_i(0)^T P_i x_i(0) + \int_0^T \nu_i^T \nu_i \; dt \tag{4.3}$$

The preparation of the above problem for SOCS is performed as described in Sections 3.2.1 and 3.2.2. We will present here the procedure and detailed equations only for nonlinear system as in Section 3.2.1. So, for this specific case, after eliminating $\overline{\nu_0}$ to get rid of algebraic constraint (4.1c), the inner optimization problem is defined as

$$\text{minimize } J = z_0(T)^p + z_1(T)^p$$

given:

$$\dot{x_0} = -x_0^2 - x_0 + v + n_0\widetilde{\nu_0} \quad (4.4a)$$

$$\dot{x_1} = -x_1^2 - 2x_1 + v + n_1\widetilde{\nu_1} \quad (4.4b)$$

$$\dot{x_2} = \dot{z_0} = (x_1 - x_0 + \overline{\nu_1})^2 + \widetilde{\nu_0}^2 \quad (4.4c)$$

$$\dot{x_3} = \dot{z_1} = \overline{\nu_1}^2 + \widetilde{\nu_1}^2 \quad (4.4d)$$

$$x_2(0) = P_0 x_0^2(0) \quad (4.4e)$$

$$x_3(0) = P_1 x_1^2(0) \quad (4.4f)$$

We are going to use Langrange-Euler form, keeping in mind that $x_2 = z_0$, $x_3 = z_1$ and that functions ϕ and L (as in Sections 3.2.1 and 3.2.2) are

$$\phi(x(0), x(T)) = x_2(T)^p + x_3(T)^p \quad (4.5a)$$

$$L(x, u, t) = 0 \quad (4.5b)$$

By including a new variable ψ (part of the cost function which measures effect of our auxiliary signal v on the system), the system of equations becomes

$$\dot{x_0} = -x_0^2 - x_0 + v + n_0\widetilde{\nu_0} \quad (4.6a)$$

$$\dot{x_1} = -x_1^2 - 2x_1 + v + n_1\widetilde{\nu_1} \quad (4.6b)$$

$$\dot{\lambda_0} = -2\lambda_2 x_0 + 2\lambda_2 x_1 + 2\lambda_2\overline{\nu_1} + \lambda_0 + 2x_0\lambda_0 \quad (4.6c)$$

$$\dot{\lambda_1} = 2\lambda_2 x_0 - 2\lambda_2 x_1 - 2\lambda_2\overline{\nu_1} + 2\lambda_1 + 2x_1\lambda_1 \quad (4.6d)$$

$$\dot{\psi} = -\psi^2 - \psi + v \quad (4.6e)$$

$$\dot{z_0} = (x_1 - x_0 + \overline{\nu_1})^2 + \widetilde{\nu_0}^2 \quad (4.6f)$$

$$\dot{z_1} = \overline{\nu_1}^2 + \widetilde{\nu_1}^2 \quad (4.6g)$$

$$0 = 2\lambda_2\widetilde{\nu_0} + n_0\lambda_0 \quad (4.6h)$$

$$0 = 2\lambda_2\overline{\nu_1} + 2\lambda_3\overline{\nu_1} - 2\lambda_2 x_0 + 2\lambda_2 x_1 \quad (4.6i)$$

$$0 = 2\lambda_3\widetilde{\nu_1} + n_1\lambda_1 \quad (4.6j)$$

with boundary conditions

$$z_0(0) - P_0 x_0^2(0) = 0 \quad (4.7a)$$

$$z_1(0) - P_1 x_1^2(0) = 0 \quad (4.7b)$$

$$\lambda_0(0) + 2P_0 x_0(0)\lambda_2 = 0 \quad (4.7c)$$

$$\lambda_1(0) + 2P_1 x_1(0)\lambda_3 = 0 \quad (4.7d)$$

$$p z_0(T)^{p-1} - \lambda_2 = 0 \quad (4.7e)$$

$$p z_1(T)^{p-1} - \lambda_3 = 0 \quad (4.7f)$$

$$\lambda_0(T) = 0 \quad (4.7g)$$

$$\lambda_1(T) = 0 \quad (4.7h)$$

$$\psi(0) = 0 \quad (4.7i)$$

$$z_0^p(T) + z_1^p(T) - 2 \cdot 1^p \geq 0 \quad (4.7j)$$

Notice the following

1. $2 \cdot 1^p$ in equation (4.7j) is there to assure that noise bound of 1 is broken

2. λ_2 and λ_3 are constants and are represented as parameters P(1) and P(2) in SOCS

The analysis of this example for various parameters will be organized in several steps.

First, we will focus on the linearized model only with L^2 norm of v as a criteria of optimality ($U = W = 0$). We will compare effect of various parameters such as noise coefficients n_i, weight on initial condition of x_i, length of testing interval T and influence of value of p in p-norm . Focus will be on comparing how far off is p-norm approach in comparison with β approach.

Then, we will focus on nonlinear model still with L^2 norm of v as our optimality criteria, and compare p-norm solution and linearized solution. We will not pay special attention to the initial guess for our variables and we will see how that affects our results.

Finally, we will focus on nonlinear model with general cost function with more in depth analysis and more sophisticated ways of setting an initial guess for our nonlinear problem.

4.1.1 Linearized Problem with L^2 Norm as a Cost Function

In this part of the analysis of Section 4.1, we are analyzing only a linearized problem with cost function of the form $\delta^2(v) = \int_0^T |v|^2 \, dt$. As pointed out in previous discussions, when our problem is linear, the p-norm is just an approximation of the optimal solution achieved by the standard β approach. In this section, the focus is on finding out how close is our p-norm approximation and how is it affected by the changes in various parameters such as noise coefficients n_i, weight on initial condition of x_i, length of testing interval T and influence of value of p in p-norm . We are also interested in the potential benefit of the p-norm approach in requiring less computational time to calculate a proper auxiliary signal v.

We utilized several values of the following parameters and their combinations:

1. Length of time interval, $T = 1$, $T = 5$ and $T = 10$
2. Noise factor in differential equations, $n_i = 1, n_i = 0.1$ and $n_i = 0.01$
3. Weight on states initial condition, $P_i = 1, P_i = 10$ and $P_i = 50$

In figure 4.1 are selected results from applying the β approach.

When we ran the same problem using p-norm approach, we obtained identical auxiliary signals v even for $p = 2$. Hence, for this simple example and for observed values of parameters, p-norm performed as well as β approach. Since each of the tests ran very briefly (1-2 seconds), it is too short a period of time to evaluate on

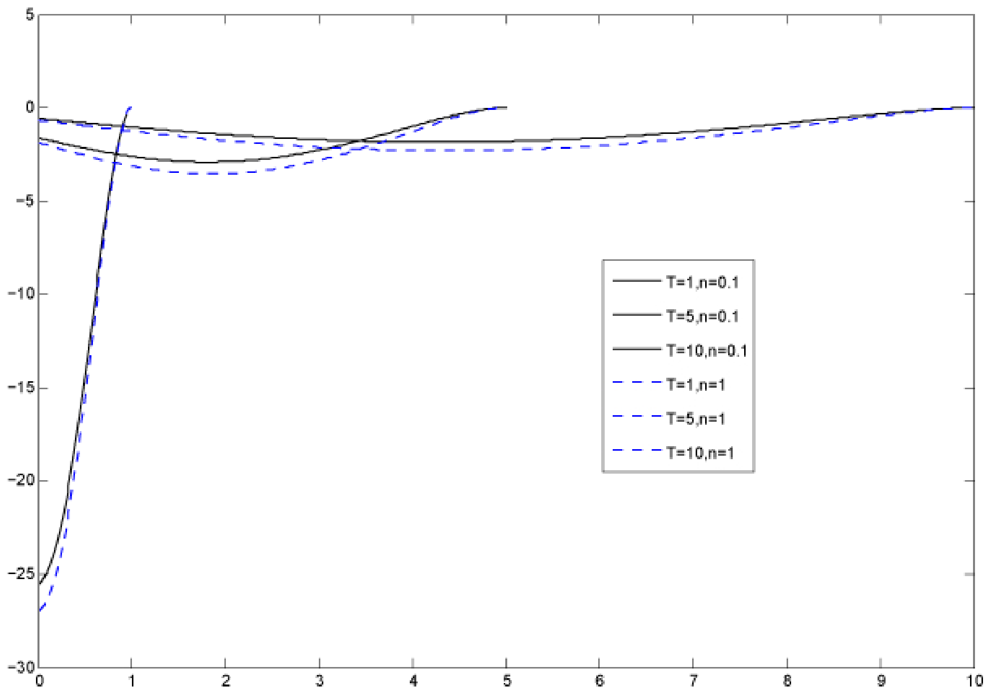

Figure 4.1: V optimal for various linearized cases using β approach, P=10

this example any differences in time performance depending on approach. We will conclude this section with analysis of parameters influence.

From our runs we were able to observe the following:

1. There was no visible difference between the auxiliary signals of the two approaches when $P_i = 10$ and $P_i = 50$ while $P_i = 1$ was very close

2. There was no visible difference between $n_i = 0.1$ and $n_i = 0.01$ cases

3. Finding auxiliary signal v was very fast in all cases (few seconds)

4. Evaluating noise bound was lasting more than a day for some instances

From the parameter analysis, we concluded that probably there is not much to gain from varying parameters P_i so we chose $P = 10$. The same is valid for parameter n_i, and we chose n_i=0.1 as our representative value. Here we should note, that if noise is much larger (order of n_i=10 and up) we might expect some differences in the results but we are not focusing on that for now. It might be part of future research.

Running time of more than a day to find noise bound was unexpected. As a direct transcription method, SOCS goes through iterations to get the solution. For examples done during my research, it typically took up to four iterations to find noise bound, and ODE error had been steadily decreased between iterations. However, in this specific example, ODE errors were increasing or oscillating for the first five to six iterations. That led to the relatively large number of iterations (seven to thirteen) which in turn increased running time since grid is refined on each iteration and the problem becomes large. We do not know what caused SOCS to behave this way, and this might be investigated further in the future.

We can see that the length of the test interval does play a significant role. Not only is our signal v larger on shorter intervals which increases the cost as seen in table 4.1, but also signal v has a different shape. The longer the interval is, the closer the shape is to the part of the sinusoid function as noticed in [7] as well. For $T = 10$ that is the case, however, we can notice that for $T = 1$ the shape is different. Hence in the future examples we will test our solutions for various values of T.

Table 4.1: Calculated cost for $n_i = 0.1$, $P_i = 10$

	T=1	T=5	T=10
Cost	261.63	22.98	17.9

4.1.2 Nonlinear Problem with L^2 Norm as a Cost Function

In this section, we are focusing on the original nonlinear problem. We are going to compare how does auxiliary signal v_{lin} found on linearized model using β approach, compare with v_p found using p-norm approach directly on the nonlinear problem. We will also present observations and some important issues when using the p-norm approach on nonlinear problems, namely importance of the initial guess.

Guided by results related to parameter relevance from Section 4.1.1, we will fix $n_i = 0.1$ and $P_i = 10$ and observe cases when $T = 1$ and $T = 5$. The obtained auxiliary signals v are presented in figures 4.2 and 4.3.

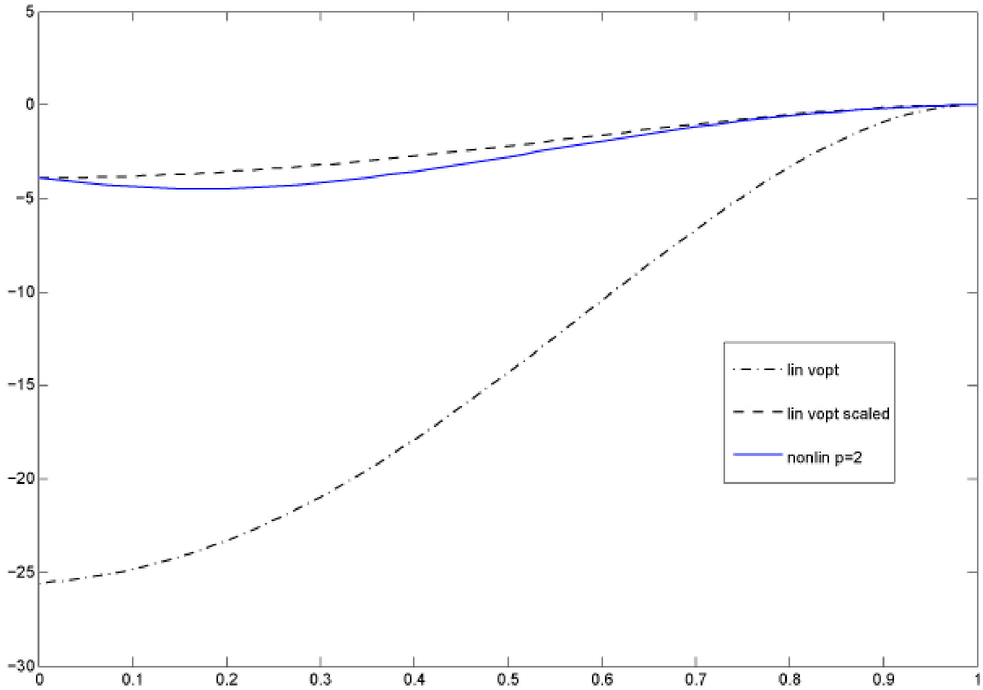

Figure 4.2: Comparison of v for T=1, n=0.1, P=10

We did runs for various values of p (p=2, 3 and 4) in the p-norm. However, we found out that the value of p for this case practically does not change the answer, so we just presented the graphs for $p = 2$. Such behavior is expected if both noise perturbations S_i (as defined in equation (4.3)) are close to the bound. This can also be seen from the figure 3.1 noticing that if S_0 and S_1 are close to bound 1, then p-norm represented by curves $S_0^p + S_1^p = 2$ is close to the max norm.

On figures 4.2 and 4.3, we also plotted scaled versions of v_{lin} so that it would be obvious that shapes of v_{lin} and v_p are different.

For this example, our cost function is just L^2 norm of auxiliary signal v. From the graphs it is quite obvious that v_p is smaller in the L^2 norm than v_{lin}. The cost

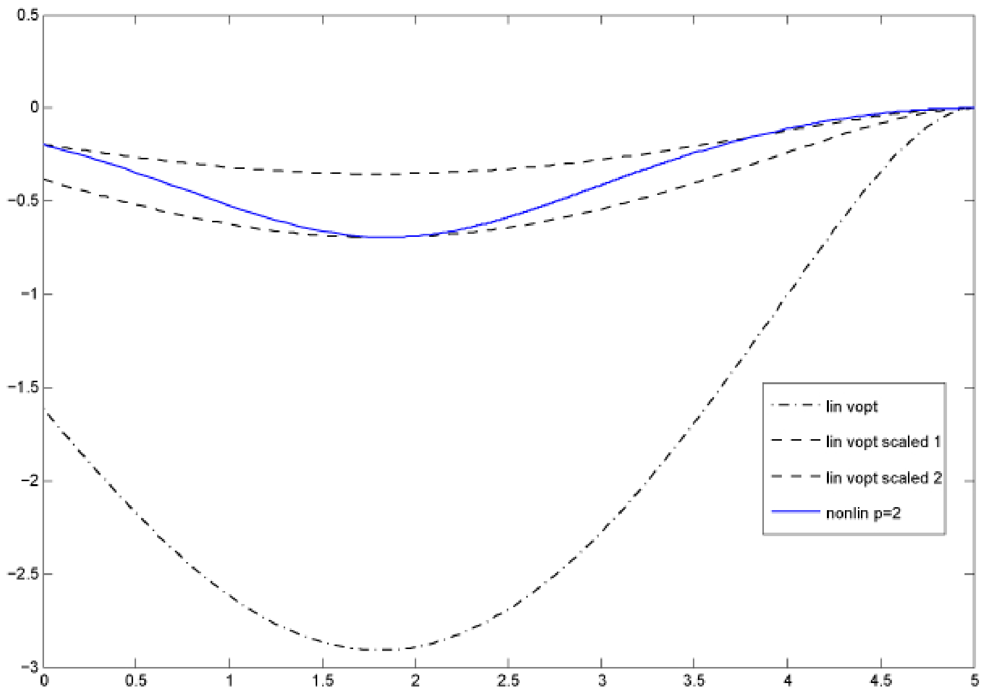

Figure 4.3: Comparison of v for T=5, n=0.1, P=10

is quantified in table 4.2. We can see that cost when using v_{lin} is significantly (over

Table 4.2: Cost of L^2 norm cost function

Parameters	Cost function		
	linearized β method	2-norm	3-norm
$T = 1$	261.64	9.01	8.97
$T = 5$	22.98	0.954	0.951

20 times) higher than the cost when applying v_p. Also, when we compare columns for cost function for $p = 2$ and $p = 3$ we can see that the difference is practically negligible. This result shows that for this case we can find significantly better auxiliary signal v using p-norm on the nonlinear problem, then if we go through the process of linearization.

We now focus on the required time needed to find auxiliary signals v. For all searches of v, Boeing's optimization software package SOCS (Section 2.3.4) was used. Code was run on PC computer with Intel Pentium M 1.6GHz processor with 760MB of RAM. SOCS traces its total running time in seconds, and those total times are enclosed in the table 4.3. For most cases, the running time was very small and quite comparable.

Table 4.3: SOCS time required to find v

Parameters	Total SOCS time in seconds		
	linearized method	2-norm	3-norm
$T = 1$	1.03 s	11.64 s	1.3 s
$T = 5$	1.64 s	1.97 s	2.08 s

We have to be very careful when reading this table. The linearized method using β approach, outperforms p-norm approach in required time. However, it is not a fair comparison. The system of linear equations is simpler than the system of nonlinear ones, so we can not draw a conclusion which approach is better with respect to time from this table.

Also, one big issue when solving nonlinear systems is exposed here: initial guess is extremely important! The required times in the table are the best times we found with respect to the random initial guesses for states and parameters. As example, for one set of initial guesses total time was 231.9 s while for another total time was 1.97 s which is presented in the table. It also explains why the required time for 2-norm and $T = 1$ in the table is higher than the other times. For some initial guesses it was even worse, SOCS was not able to find the solution! The typical problems were that SOCS was not able to find a feasible point, solution was not converging, or we would just get some completely useless results. Hence the human time was significantly higher in finding p-norm solutions due to the nonlinearity of the problem. It is not too crucial for practical applications since v is usually calculated off-line and ahead of time, but we can do better as shown later.

We would like to point out, that switching from default discretization-integration method (trapezoidal rule) to Hermite-Simpson was important in obtaining auxiliary signals v_p in some instances. When left to SOCS to switch from trapezoidal to Hermite-Simpson after a few iterations, often a feasible point was not found. By starting with Hermite-Simpson, that was avoided. Also, starting Hermite-Simpson helped reduce the running time. This might be just the specific property of this example, but the effect of discretization-integration method might be worth exploring in future work.

Finally, we used the algorithm from Section 3.4 to evaluate what is the maximum allowed amount of noise in the system when applying the obtained v_{lin} and v_p. When

Table 4.4: Noise bounds when using v_{lin} and v_p

Parameters	Noise bound		
	linearized method	2-norm	3-norm
$T = 1$	5579.2	1.038	1.025
$T = 5$	1293.5	1.035	1.022

we were finding auxiliary signal, our request was that noise bound is greater than 1. The results from the table 4.4 when applying linearized signal v_{lin} are quite high. This result implies that, though our v_{lin} is proper, it is far from optimal. One explanation for this is that the nonlinearity is helping with fault detection so a smaller signal is needed in the nonlinear problem.

Since the calculated noise bounds are significantly larger than 1, it is logical to expect that we can get a better auxiliary signals v (in L^2 norm sense). We looked for new auxiliary signal $v_2 = k * v_1$ where v_1 was the calculated auxiliary signal. By trial and error, we found the smallest k such that v_2 is still proper when applied to the nonlinear problem. Considering the case with parameters $T = 5, n_i = 0.1$ and $P_i = 10$, we found that $k = 0.21$ is still providing us with proper signal when v_1 is v_{lin}.

However, we should keep in mind that not only are we using random initial guesses to find auxiliary signals, but now we were using random initial guesses in the code that was finding noise bound for nonlinear system as well. Also, we are using necessary conditions to set up system of equations. For nonlinear systems, this set of equations is not representing a sufficient condition for global minimum. This could lead to few possible explanations as to why such a high noise bound occurred:

1. Existence of local minimums while searching for optimal auxiliary signals and SOCS getting stuck in one of the local minimums

2. Existence of local minimums while searching for noise bound and SOCS getting stuck in one of the local minimums

3. Finding local maximum of inner problem instead of minimum in both cases since the necessary conditions for local maximum are the same as necessary conditions for local minimum

4. Existence of local minimums in inner optimization problem

To avoid some of these issues, a better way of choosing initial guess will be required.

In this subsection, we found that the p-norm approach on nonlinear problem found significantly better auxiliary signals than those obtained using β approach on linearized model. However, the importance of good way of choosing initial guess presented itself. Just using random initial guess, human time required was significant, SOCS running time was significant, and it might be the cause of obtaining higher noise bounds than they should be. This issue prodded us to find more sophisticated ways of choosing an initial guess, and some ideas are in the following subsection.

4.1.3 Nonlinear Problem with General Cost Function

So far, we were trying to minimize L^2 norm of auxiliary signal v. We observed linearized and nonlinear version. When problem was nonlinear, we found that v_p outperforms v_{lin}, but that a more sophisticated method of determining an initial guess is needed. In this section, we will focus on different cost functions and examine their influence for nonlinear and linearized problem. We will also present the ways of choosing the initial guess for nonlinear cases which turns out to be crucial in finding new results.

Let the cost function be defined as

$$\delta^2(v) = \psi(T)^T W \psi(T) + \int_0^T |v|^2 + \psi^T U \psi \; dt \tag{4.8}$$

where at least one of the matrices U or W is a non zero positive semidefinite matrix. The new variable ψ is used to measure the impact our auxiliary signal v has on the system. It is defined as

$$\dot{\psi} = f(\psi, v), \;\; \psi(0) = 0 \tag{4.9}$$

Though $f(\psi, v)$ can be any function, in this thesis we are assuming that we want to minimize effect on the non-faulty system, hence ψ and x_0 will have the same dynamics without noise, that is, $f(\psi, v) = f_0(\psi_0, v)$.

Since ψ is not part of what we called the inner minimization problem, the effect of this new cost function on our system of equations is minimal. Beside modifying the cost function itself, we just have an additional system of differential equations and a new boundary condition (4.9) as a new constraints. Let's repeat the definition of the problem:

$$\dot{x_0} = -x_0^2 - x_0 + v + 0.1\widetilde{\nu_0} \tag{4.10a}$$

$$\dot{x_1} = -x_1^2 - 2x_1 + v + 0.1\widetilde{\nu_1} \tag{4.10b}$$

$$\dot{\psi} = -\psi^2 - \psi + v \tag{4.10c}$$

$$0 = x_1 - x_0 + \overline{\nu_1} - \overline{\nu_0} \tag{4.10d}$$

with linearized system being

$$\dot{x_0} = -x_0 + v + 0.1\widetilde{\nu_0} \tag{4.11a}$$

$$\dot{x_1} = -2x_1 + v + 0.1\widetilde{\nu_1} \tag{4.11b}$$

$$\dot{\psi} = -\psi + v \tag{4.11c}$$

$$0 = x_1 - x_0 + \overline{\nu_1} - \overline{\nu_0} \tag{4.11d}$$

We are going to consider 2 values for the fixed final time $T = 10$ and $T = 1$, an additive noise bound of 1, and weight on the initial condition of states x_0 and x_1 of $P = 10$. Also, for now, we will assume that $W = 0$. Thus we reduce the overall response but do not emphasize the value of ψ at the end of the test.

Before proceeding to the results, we would like to discuss the strategies we used to find new initial guesses for the nonlinear case.

1. Using solution of the linearized example as initial guess for the nonlinear problem

2. Finding solutions of simplified inner nonlinear optimization problem and using it as an initial guess

It was shown [7] that the linearized problem using β approach provides us with the optimal solution of (4.11). While it does not have to be (and turn out it is not) the optimal solution for original nonlinear problem (4.10), it does provide us with something to start with, an initial guess. We used this approach in some of the examples. For this specific example however, a different strategy gave us better results that are presented in this section.

The idea is to simplify the inner nonlinear optimization problem to provide SOCS an easier task of finding a feasible point. We noticed that we have the crucial constraint outside of the inner optimization problem that assures that v is proper of the form

$$z_0{}^p(T) + z_1{}^p(T) \geq 2 \tag{4.12}$$

where z_i's are measures of additive noise in the system defined as

$$\begin{aligned}
\dot{z}_0 &= \|\overline{\nu_0}\|_2^2 + \|\widetilde{\nu_0}\|_2^2 && \text{(4.13a)}\\
\dot{z}_1 &= \|\overline{\nu_1}\|_2^2 + \|\widetilde{\nu_1}\|_2^2 && \text{(4.13b)}\\
z_0(0) &= x_0(0)^T P_0 x_0(0) && \text{(4.13c)}\\
z_1(0) &= x_1(0)^T P_1 x_1(0) && \text{(4.13d)}
\end{aligned}$$

That inspired us that simplification of dynamics of z_i can help SOCS significantly, so we defined new dynamics for z_0 by eliminating one of the noise components

$$\dot{z}_0 = \|\overline{\nu_0}\|_2^2 \tag{4.14}$$

All other equations in inner optimization problem remained the same, and included omitted noise component $\|\widetilde{\nu_0}\|_2^2$.

Solution obtained this way, gave us an initial condition that we used on the original nonlinear problem, and obtained one of the results that will be presented (case when $T = 10$ and $U = 30$). Once obtaining that solution of the original nonlinear problem, we used that solution as an initial guess for all other runs for different values of U and T. The analysis of this simplification method, why exactly it worked and the insight into what kind of simplification would be the most beneficial could be part of the future research.

Now we are ready to present our results.

First we will observe the effect of U on the auxiliary signal v for the linearized system. For $U = 0, 1, 10$ and 30 the results are presented on figure 4.4. From the figure and cost table 4.5 we can see that as U increases, v is gradually growing and the cost is increasing. There is barely noticeable change in v when going from $U = 0$ (this is just L^2 norm) to $U = 1$. For the same parameters, just applied on the

Table 4.5: Cost of linearized model for various values of U

Parameter U	U=0	U=1	U=10	U=30
Cost	261.6	310	724	1575
L^2 norm of v	261.6	262.0	281	333

nonlinear system, the results are on figure 4.5 and in table 4.6. From tables 4.5

Table 4.6: Cost of nonlinear model for various values of U

Parameter U	U=0	U=1	U=10	U=30
Cost	9	43	241	606
L^2 norm of v	9	16	41	97

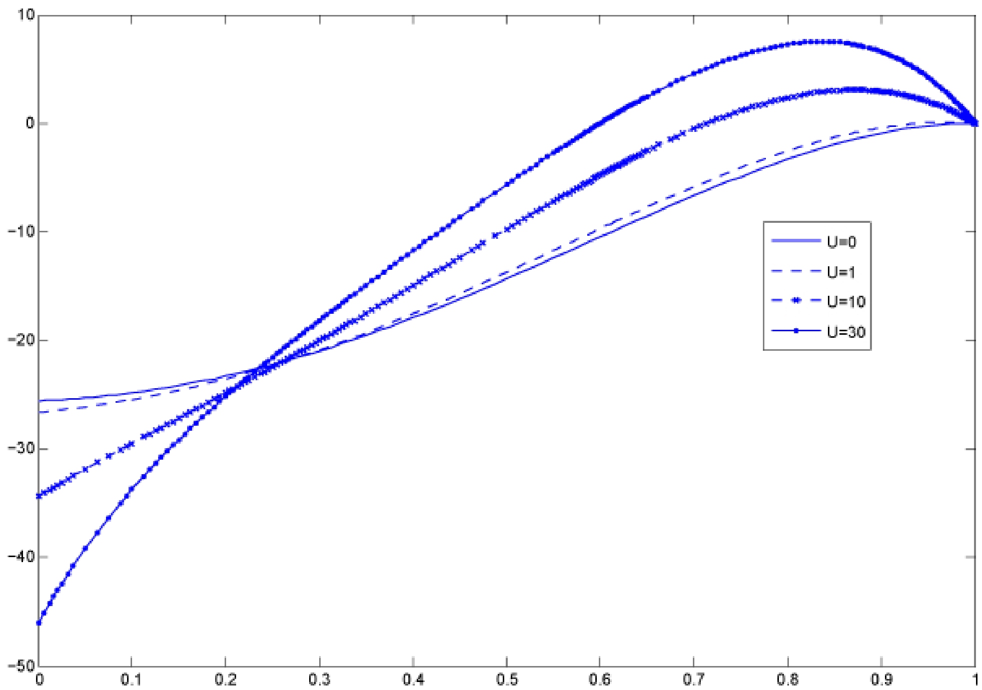

Figure 4.4: Effect of U on v for linearized model

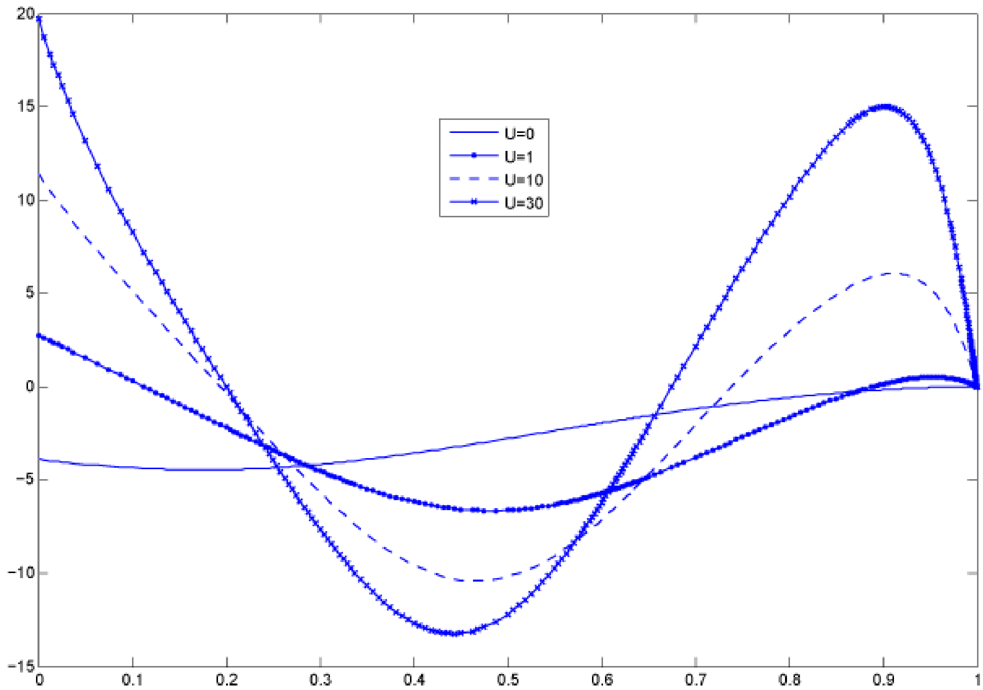

Figure 4.5: Effect of U on v for nonlinear model

and 4.6, noticing the influence of L^2 norm of v on total cost, we can observe that the influence of ψ (through matrix U) is much stronger for nonlinear cases than for linearized ones.

Unlike for the linearized case, here the change from $U = 0$ to $U = 1$ corresponds to the significant increase in cost, and noticeable change in v. Though the shape of the linearized signal v_{lin} and v_p were different for $U = 0$, they had some similarities. When $U = 10$ the difference is much more pronounced as shown on figure 4.6. The

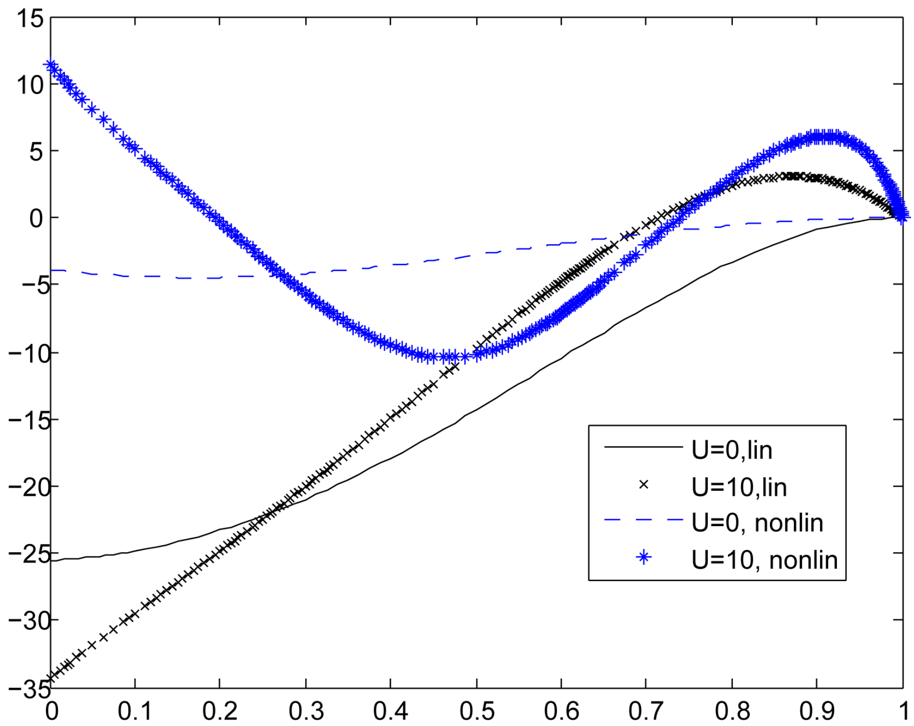

Figure 4.6: Comparison between v_p and v_{lin} for $U = 0$ and $U = 10$

reason for such significant difference can be found by a brief analysis of differential equations for ψ for linearized and nonlinear model.

For the linearized model, there is only one equilibrium at $\psi = 0$ and it is of the sink type. However, for nonlinear model we have 2 equilibria, one of the sink type at $\psi = 0$ but also one of the source type at $\psi = -1$ as shown in the figure 4.7. We presented equilibria only for nonfaulty case since dynamics of ψ are following nonfaulty model. There is difference in equilibria for faulty model as well. When $U = 0$, we are not

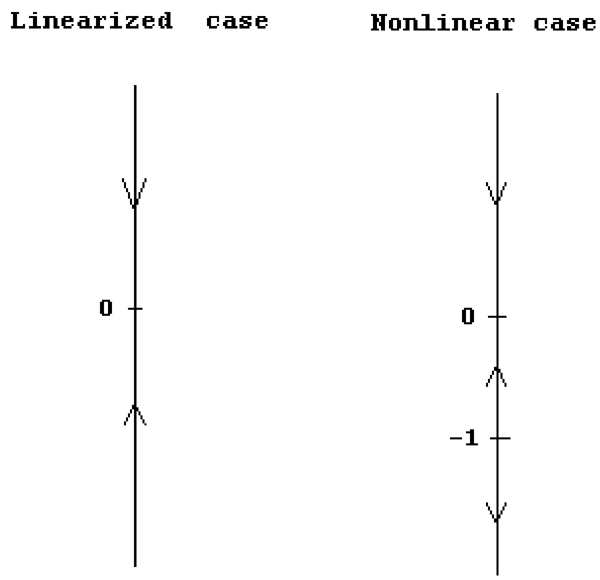

Figure 4.7: Equilibrium analysis of linearized and nonlinear nonfaulty model

affected by this difference since cost does not depend on ψ. However, as soon as ψ started playing role in our cost function, the fact that nonlinear system has source equilibrium that linearized is lacking makes a huge difference. Figure 4.8 shows ψ for calculated v_p and v_{lin} when $U = 1$. The end behavior of ψ near T shows the effect of equilibria. Near T, our auxiliary signal is getting smaller. If ψ happens to be less than -1 when v is small enough, equilibria for nonlinear case will cause it to decline very fast while linearized case will assume that ψ will go toward 0. The effect is even more pronounced when U is larger. This is showing that extreme caution has to be taken if we want to use linearized approach for cost function that has $U \neq 0$.

We would like to point out the upward bump on the graph 4.5 in signals v near

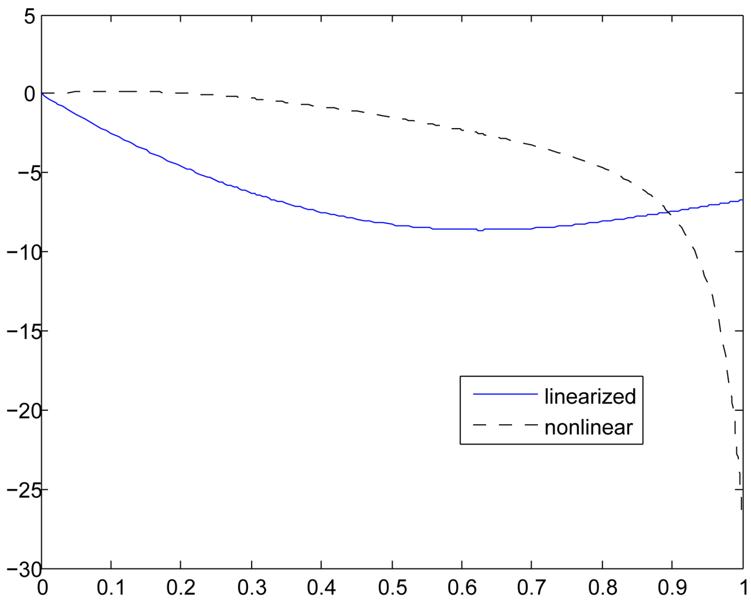

Figure 4.8: Comparison of signal ψ for linearized and nonlinearized models when $U = 1$

the very end of the interval. It is due to the fact that SOCS wanted the ψ value near the end of the time interval to get closer to 0, and that way to make cost increase by ψ when it shoots down (when v gets close to 0 near the end of the interval) minimal.

We also observed the behavior for $T = 10$. The results when applying p-norm on the nonlinear problem are in the figure 4.9. Notice the characteristic bump in v near the end of the time interval that we observed for $T = 1$ case. As expected, the stronger the effect of ψ is (U being greater), the stronger the bump has to be.

For linear problems with simple cost function (L^2 norm of v) it was observed [7] that as time interval increases, periodicity of the signal v occurs. We can see that it seems to be the case for nonlinear problems with more complex cost function as well. The cost function and SOCS running time are in the table 4.7.

We can notice that cost is growing as expected. Also, noise bounds are quite close to our desired value of 1 implying that our calculated signals v_p are not far off from the optimal ones. Unlike the previous examples where random choice of initial

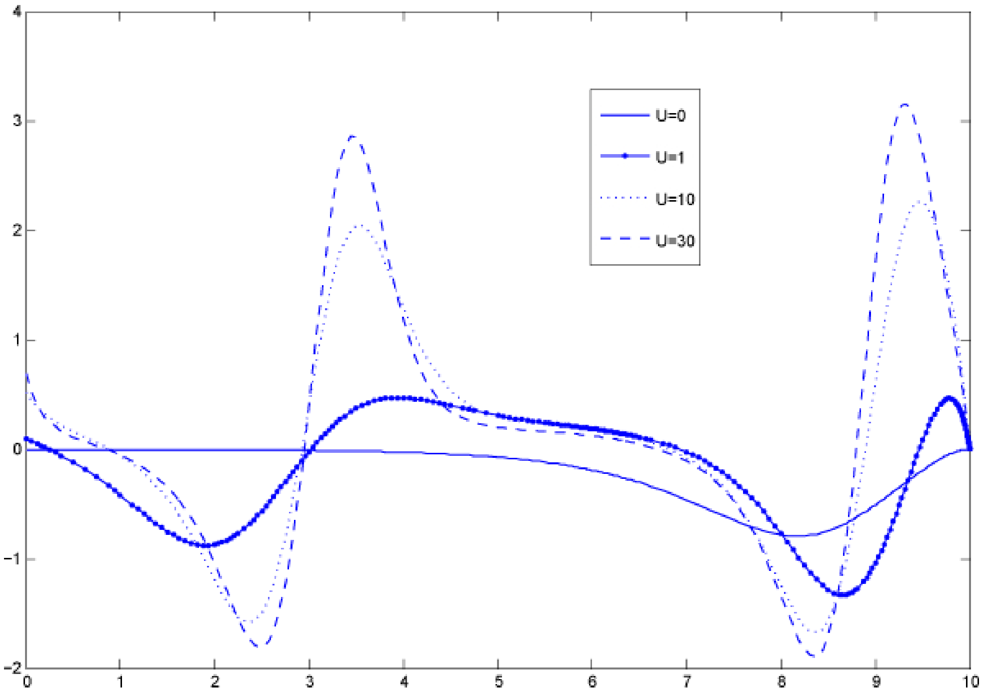

Figure 4.9: Test signal v on $[0, 10]$ for different U values

Table 4.7: Algorithm performances for $T = 10$ and various values of U

Parameter U	U=0	U=1	U=10	U=30
Cost	1.12	11.75	77.21	207.3
Time(sec)	54.5	12.0	11.1	17.4
Noise bound	1.018	1.005	1.004	1.003

guess was requiring significant human time to find the result, using more a structural approach to the initial guess made human time involved minimal.

In this subsection, we combined more complex cost function with a nonlinear problem. To avoid significant human time in finding a good initial guess, 2 methods are proposed. They worked quite well on this example, providing us in little computational time the solutions. The results pointed out the weakness in applying the linearized signal when having ψ in the cost function resulting from the change of the equilibria of the system.

Notice, that each subsection can be observed as an example on it's own. Overall, we started by analyzing a linear problem with simple cost function, and used it to confirm that p-norm is performing similar to β norm approach for linear problems. We also identified parameters whose change significantly influences auxiliary signal v.

Then we introduced nonlinearities and kept the cost function simple. We noted several potential problems with the p-norm approach. Namely the influence of the initial guess on performance of the search by significantly increasing human time involved and resulting in sub par performances. We also pointed out potential pitfalls like local minimums. It was shown that v_p outperformed v_{lin} by having significantly lower cost. We also presented the way of improving v_{lin} by scaling it.

Finally, we observed nonlinear system with more complex cost function. By using 2 methods to find a good initial guess, we eliminated human time issue. We showed that linearization is not a good way to go, and we presented why. We obtained nearly optimal auxiliary signals v, for 2 different lengths of testing interval.

4.2 Example where Linearization does not Work

This is a simple theoretical example where linearization would not give us any result. The problem is defined as

$$\text{minimize} \int_0^T |v|^2 \; dt$$

given

$$\dot{x_0} = -x_0^2 + v + \widetilde{\nu_0} \tag{4.15a}$$

$$\dot{x_1} = -2x_1^2 + v + \widetilde{\nu_1} \tag{4.15b}$$

$$0 = x_1 - x_0 + \overline{\nu_1} - \overline{\nu_0} \tag{4.15c}$$

Matrix $P = 1$ in $x_i(0)^T P x_i(0)$. Using my p-method code, I obtained the following costs for various values of p:

- $p = 2$ cost is 7.56, noise bound 1.18
- $p = 3$ cost is 7.37
- $p = 4$ cost is 7.26

The proper signal v found is in the figure 4.10. It is interesting to note that model linearized about $x_i = 0$ of this problem can not provide the solution since the linearized dynamics of both systems, x_0 and x_1, will be the same.

4.3 Real Physical System Example

This problem is real life problem [47] which describes motion of the pendulum with fault manifested by a change in the friction coefficient in the joint. The model was used in [47] as an example where linearizations are expected to work well. The nonlinearity is of the sinusoidal type, hence it has a high probability of multiple local minimums and maximums. The models obtained through physics of the system are:

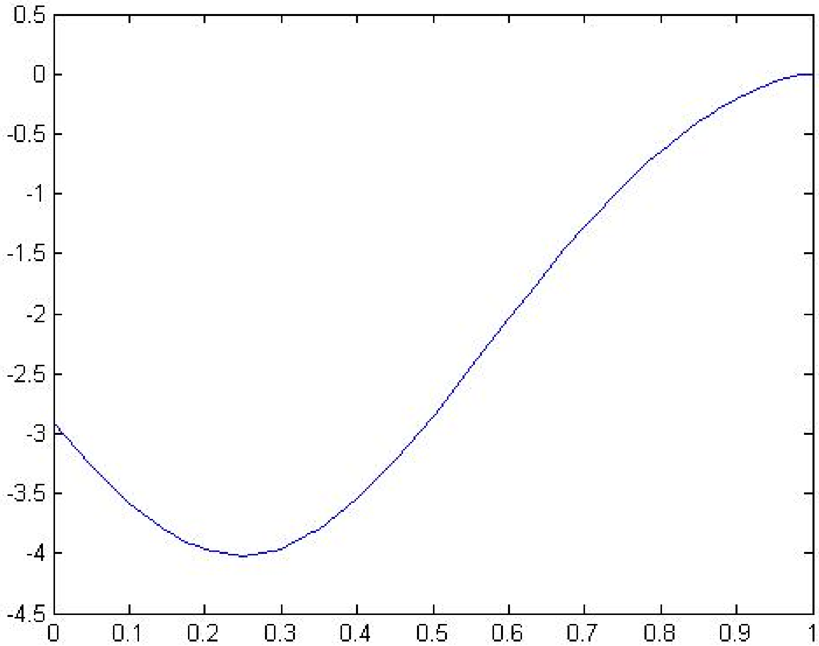

Figure 4.10: Proper signal v for Example 4.2 found using 2-norm

Nonfaulty nonlinear model

$$x_1' = x_2 \tag{4.16a}$$
$$x_2' = -9\sin x_1 - \frac{1}{5}x_2 + u + 0.5\mu_{01} \tag{4.16b}$$
$$y = x_1 + 0.25\mu_{02} \tag{4.16c}$$

Faulty nonlinear model

$$x_1' = x_2 \tag{4.17a}$$
$$x_2' = -9\sin x_1 - x_2 + u + 0.5\mu_{11} \tag{4.17b}$$
$$y = x_1 + 0.25\mu_{12} \tag{4.17c}$$

The linearization of (4.16) and (4.17) about the equilibria (0,0) yields:

$$x' = \begin{bmatrix} 0 & 1 \\ -9 & -\frac{1}{5} \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u + \begin{bmatrix} 0 & 0 \\ 0 & 0.5 \end{bmatrix} \mu \tag{4.18a}$$
$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} x + \begin{bmatrix} 0.25 & 0 \end{bmatrix} \mu \tag{4.18b}$$

and

$$x' = \begin{bmatrix} 0 & 1 \\ -9 & -1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u + \begin{bmatrix} 0 & 0 \\ 0 & 0.5 \end{bmatrix} \mu \tag{4.19a}$$
$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} x + \begin{bmatrix} 0.25 & 0 \end{bmatrix} \mu \tag{4.19b}$$

respectively.

Our test interval is $[0, 5]$, weight on initial values $x_i(0)$ is $P = I$ and objective function is

$$\text{minimize } \delta^2(v) = \int_0^5 |v|^2 \, dt$$

In this example, we used the solution obtained by p-norm approach on the linearized system v_{lin} as an initial guess for the nonlinear solver. The calculated v_{lin} and

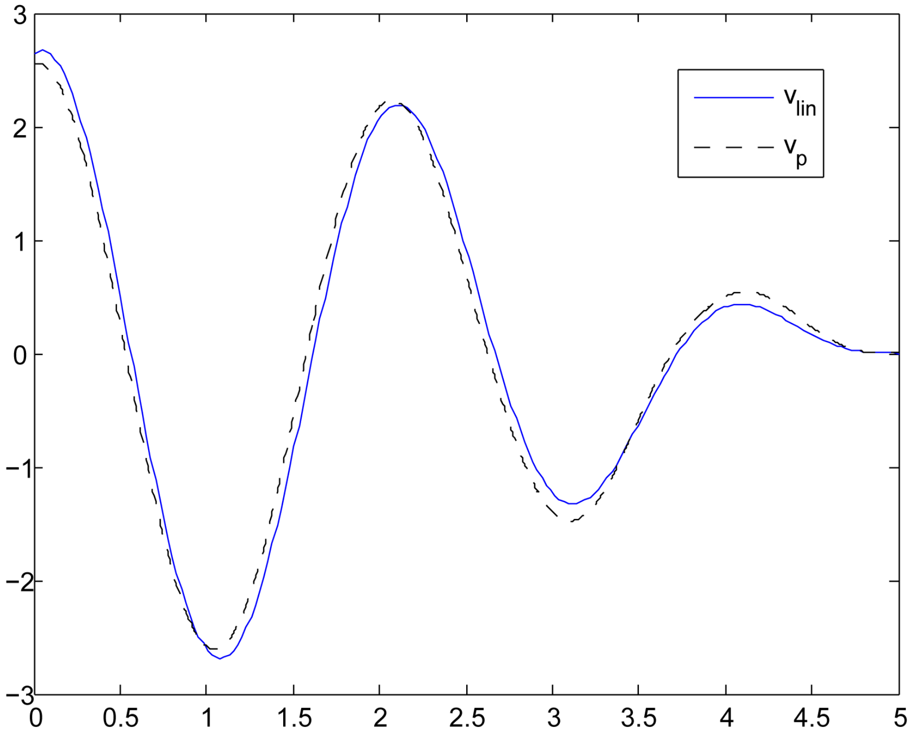

Figure 4.11: Comparison of v_{lin} and v_p for Example 4.3

Table 4.8: Performance comparison

Method	Cost function	Time	Noise bound
linearized	9.413	5.77	1.032
2-norm	9.131	7.67	1.008

v_p are presented in figure 4.11 and the relevant performance quantifiers in table 4.8. From the table we can see that v_p obtained by p-norm approach ($p = 2$) outperformed v_{lin}, but the cost difference is slight. Noticing that Taylor series of sinusoidal function does not have quadratic term, linear approximation is quite good. Hence such a small improvement by using p-norm method. It took more time to calculate v_p, but the time was comparable. Both signals were proper.

This example presents situation where there is not much gain by solving the problem as nonlinear.

However, it is a good example for pointing out again the importance of the initial guess when dealing with nonlinear problems. This time we will analyze effects when trying to evaluate the noise bound. In this specific example, there are a lot of local minimums since a sinusoidal function is involved. This problem was linearized about $(x_0, x_1) = (0, 0)$. However this problem has multiple equilibria of the type $(k\pi, 0), k \in Z$ and the results highly depend on whether our initial guess is near the linearization point or not.

In the following table were noise bounds found for different values of initial guesses for variables. We were keeping initial guess for noise measure $P = 100$, initial number of grid points in SOCS is $NGRID = 81$, and trapezoidal discretization method for all cases. The initial guess for variable Y is the linear approximation for given $Y(0)$ and $Y(T)$. The results in table 4.9 clearly show that choice of initial guess close

Table 4.9: Noise bound for various initial guesses

$Y(0)$	$Y(T)$	Time(s)	Noise bound
0	0.1	119	1.008
10	100	1278	12.44
-10	-100	41878	84.31
3.2	3.5	1142	8.3
-3.2	-3.5	359	9.94
-10	10	340	1.008

to different equilibria produces different noise bounds. Initial guess (3.2,3.5) was targeting equilibria $(\pi, 0)$ and it affected result by allowing more noise in the system. Also we can notice that since the problem is not symmetric, a change of sign in the initial guess (-3.2,-3.5) gave a different noise bound than (3.2,3.5). The extraordinary long time for (-10,-100) might be caused from having multiple minimum within the range, and huge difference between its noise bound and noise bound of initial guess (10,100) might be from SOCS choosing different corresponding minimums.

4.4 Model Uncertainty Example

This example is a modification of the Example 4.2.3 from [7] where the effect of weight on the initial states P on the shape of the auxiliary signal v was presented. In [7] it was solved using the β method with Riccati equations. Here we test our new direct optimization code.

We focused on 2 values of P, $P = I$ and $P = 0.0001I$ that were shown to produce distinctive shapes for v, to investigate the effect of the amount of model uncertainty on the shape of the auxiliary signal v. We are going to use the direct optimization approach as described in Section 3.3.

The optimization goal in this example is to

$$\text{minimize } \delta^2(v) = \int_0^1 |v|^2 \, dt$$

where our nonfaulty and faulty linear models are defined as:

$$\dot{x_0} = (-2 + g_0\delta_1)x_0 + v + \begin{bmatrix} 1 & 0 \end{bmatrix} \nu_0 \tag{4.20a}$$

$$\dot{x_1} = (-1 + g_1\delta_2)x_1 + v + \begin{bmatrix} 1 & 0 \end{bmatrix} \nu_1 \tag{4.20b}$$

$$0 = x_0 - x_1 + \begin{bmatrix} 0 & 1 \end{bmatrix} \nu_0 - \begin{bmatrix} 0 & 1 \end{bmatrix} \nu_1 \tag{4.20c}$$

with noise perturbation and involved matrices and variables defined as in (3.33)

$$S_i = x_i(0)^T P_i x_i(0) + \int_0^{s_i} \varphi_i^T \Gamma \varphi_i \, dt \tag{4.21}$$

Our testing time interval is $t \in [0, 1]$.

For $P = I$, the auxiliary signal v for various values of weight on model uncertainty $g = g_0 = g_1$ are in the figure 4.12 and relevant running parameters are in the table 4.10. In all of these runs except $g = 0.2$, we left the final time s to be free with

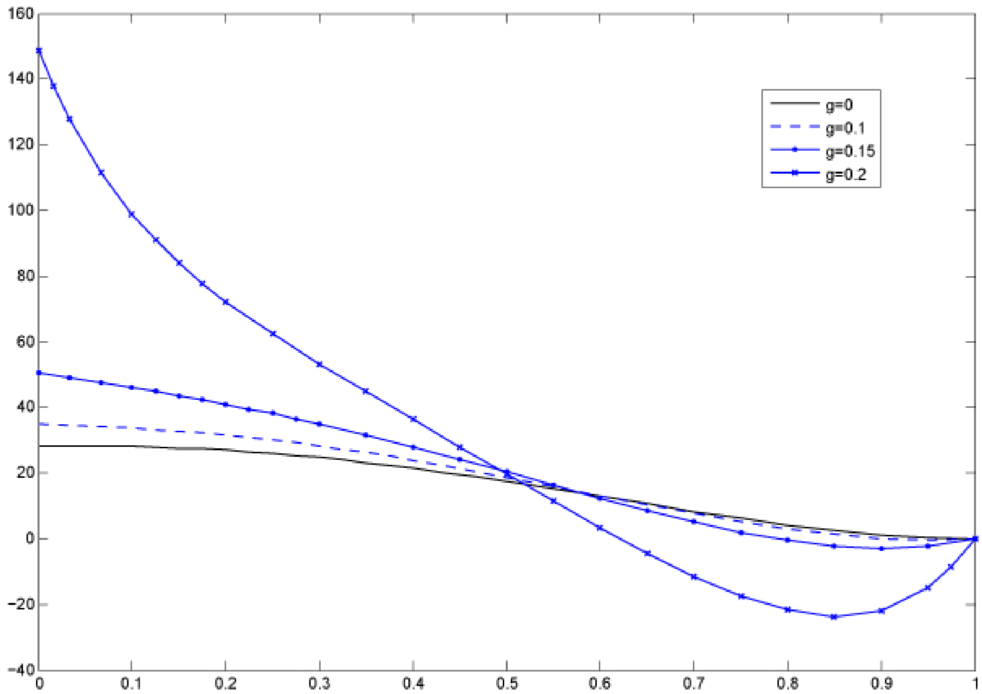

Figure 4.12: Auxiliary signal v for various values of g, and $P = I$

Table 4.10: Parameters for various g and $P = I$

g	β	Cost	Time(s)
0	0.476	353.7	0.55
0.1	0.495	468.7	1.09
0.15	0.516	763.9	1.19
0.2	0.539	3005	1.05

initial guess of $s = 0.1$. In all cases we got at the end, that for optimal proper v the best value of s is $s = 1$. For case $g = 0.2$ we had problems with SOCS finding feasible point for initial guess $s = 0.1$, so we used initial guess $s = 1$ to obtain v. We let SOCS pick initial guesses for all other variables.

We can notice that the shape of v highly depends on g. While the difference in shapes of v between cases when $g = 0.1$ and $g = 0.15$ is not huge, going to $g = 0.2$ creates the quite different shape and huge increase in cost. The possible reason is that the effect of model noise depends on the size of states x as well.

As can be seen from the table 4.10, the required running time was very small.

We did the similar analysis for $P = 0.0001I$ with auxiliary signals v for various g presented in figure 4.13 and relevant parameters are presented in table 4.11. In this

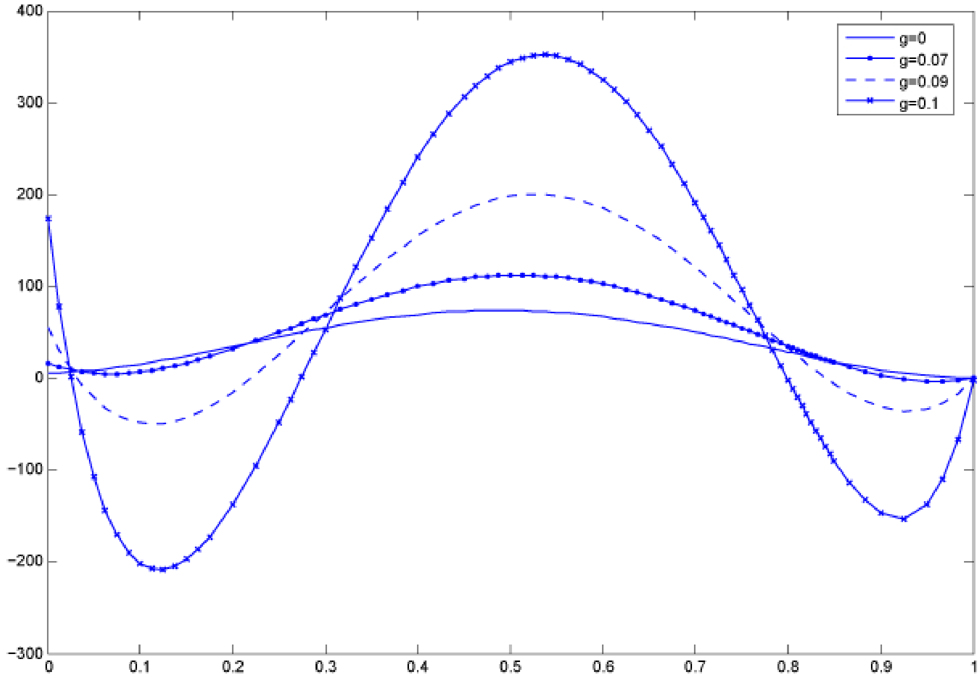

Figure 4.13: Auxiliary signal v for various values of g, and $P = 0.0001I$

Table 4.11: Parameters for various g and $P = 0.0001I$

g	β	Cost	Time(s)
0	0.501	2228.1	3.56
0.07	0.505	4516.8	2.00
0.09	0.506	12139	2.92
0.1	0.507	40283	3.52

case, for the initial guess of all values of $g \neq 0$ we just used the results from the $g = 0$

run.

In comparison with the previous $P = I$ case, we can observe significantly higher costs. It is not surprising, since the more weight we have on the initial value of the states, the more they will dominate the noise perturbation expression. When P was very small, like in this case, noise components became more dominant. That is the reason why not only we have higher cost, but it is achieved for smaller values of g and the shape of the auxiliary signal v is getting noticeably changed as soon as g becomes different than 0. Again, the time to obtain v was very short. We can also notice that the value of β did not change much with the increase of g.

4.5 Other Extensions of Linear Model Uncertainty Examples

As pointed out in Section 3.3, there was a way to attack certain model uncertainty problems using Riccati equations. In this section, we will present cases where Riccati approach does not work, and the global optimization approach does.

All extensions will be done on the special case of the previous Example 4.4, where we will observe the situation when there is no weight on initial value of states x, i.e. $P = 0$. We would like to point out that Riccati approach was not done for this case. The setup is similar to the one in the previous example:

$$\text{minimize } \delta^2(v) = \int_0^1 |v|^2 \, dt$$

where our nonfaulty and faulty linear models are defined as:

$$\dot{x_0} = (-2 + g_0\delta_1)x_0 + v + \begin{bmatrix} 1 & 0 \end{bmatrix} \nu_0 \tag{4.22a}$$

$$\dot{x_1} = (-1 + g_1\delta_2)x_1 + v + \begin{bmatrix} 1 & 0 \end{bmatrix} \nu_1 \tag{4.22b}$$

$$0 = x_0 - x_1 + \begin{bmatrix} 0 & 1 \end{bmatrix} \nu_0 - \begin{bmatrix} 0 & 1 \end{bmatrix} \nu_1 \tag{4.22c}$$

with noise perturbation and involved matrices and variables defined as in (3.33)

$$S_i = \int_0^{s_i} \varphi_i^T \Gamma \varphi_i \, dt \tag{4.23}$$

Our testing time interval is $t \in [0, 1]$. The auxiliary signal v obtained for various values of g is in figure 4.14 with parameters presented in table 4.12. Note that increasing g means we are increasing the amount of model uncertainty in the problem.

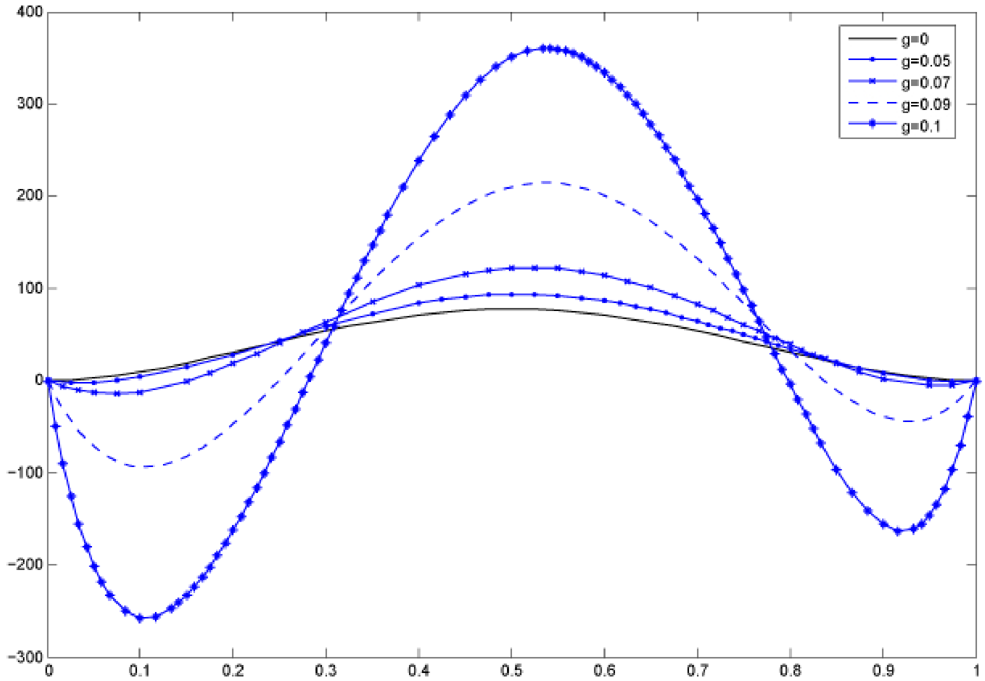

Figure 4.14: Auxiliary signal v for various values of g, and $P = 0$

Table 4.12: Parameters for various g and $P = 0$

g	β	Cost	Time(s)
0	0.499	2381.3	8.28
0.05	0.501	3285.2	785.9
0.07	0.502	5122.3	2.65
0.09	0.505	14310	2076.2
0.1	0.507	45555	21.73

When we compare graphs 4.14 and 4.13 and tables 4.12 and 4.11, we can observe a lot of similarities. One glaring difference though, is the running time. As pointed out in Chapter 4.2.1 of [7], having $P \neq 0$, no matter how small will guarantee that our output sets are bounded and convex. We can not guarantee boundedness and convexness for $P = 0$, which in return increases the complexity of the problem for numerical solvers. Hence the increase and variety of the running times for this example.

4.5.1 Hard Bound on the Auxiliary Signal v

In a lot of systems we can not have unlimited size of the auxiliary signal, hence we have some constraints on our auxiliary signal v. In this subsection we will observe so called hard bounds, where we limit our value of v to be in the form of $v_{min} \leq v \leq v_{max}$ or $v \leq v_{max}$. These problems can not be done by β approach using Riccati equations of [7]. The implementation in SOCS is quite straight forward, since SOCS is providing the way of implementing that feature.

First, we observed the case when $g = 0$, i.e. the case without model uncertainty. On the graph 4.15 we can see effects of the specific bounds. It was shown [7] that for linear cases if v is proper, so is $-v$. While SOCS was opting for v that has positive values for cases without constraints on v and $|v| \leq 60$, for the case when $v \leq 50$, SOCS was able to properly choose signal that is the same as $-v$ for the non constrained problem.

When picking the hard bound, we have to be careful that our pick makes sense. It is easily possible to set constraints on v that produce no solutions. That would be the case for this example if we set $|v| \leq 50$ and lower.

Now, we will observe the case when model uncertainty is present with the model uncertainty coefficient $g = 0.05$. We will use the solution without v constraints as an initial guess, and we will leave the ending time free. The results are in the figure 4.16. Notice that for a bound of the type $v_{min} \leq v \leq v_{max}$, we have both v_{min} and v_{max} playing an important role as shown by the $|v| \leq 71.2$ case. That is not surprising since the shape of the auxiliary signal when model uncertainty is present is more complex (as shown on figure 4.14). Also, probably due to the increased difficulty, this

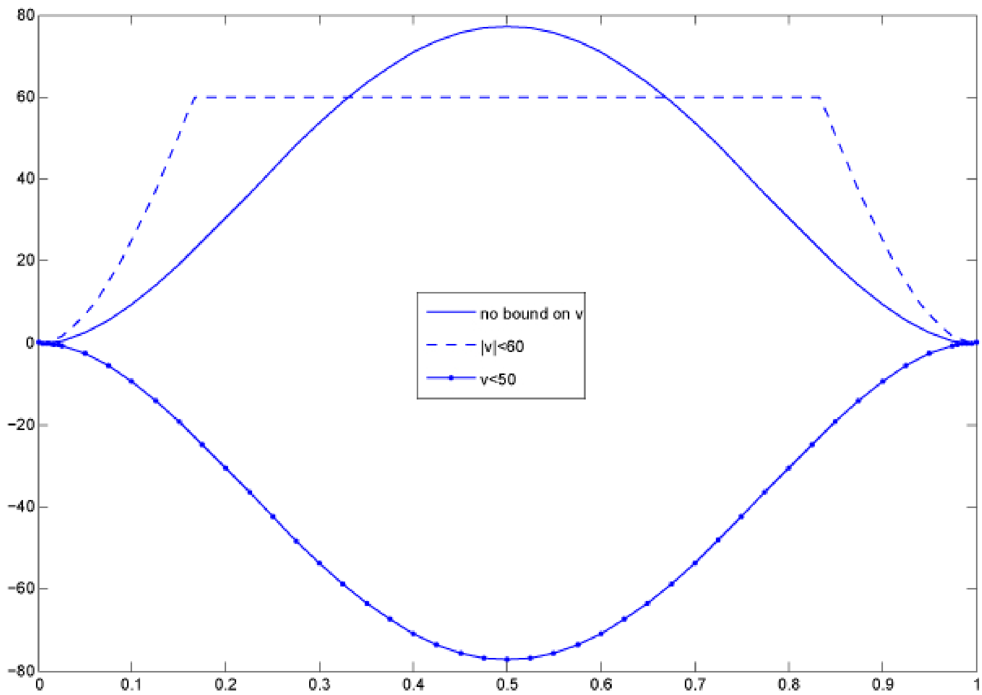

Figure 4.15: v with hard bound on auxiliary signal v, $g = 0$

time SOCS was not able to find that $-v$ is the solution of $v \leq v_{max}$ type of bound when v was an initial guess, since it was not able to find feasible point. To obtain result for this case, we used $-v$ for the initial guess.

4.5.2 Soft Bound on the Auxiliary Signal v

Sometimes, we do not want to be so strict on the constraint on v to set up hard bounds. We would just like to discourage v from approaching certain values. That can be achieved by increasing the cost function if v approaches the specific value (or trajectory). A common cost function used so far was just the L^2 norm of the auxiliary signal v. To use soft bounds, we choose 2 new cost functions

1. Soft bound:

$$\text{minimize } \delta^2(v) = \int_0^1 \frac{c_s}{(r^2 - v^2)^2} \, dt$$

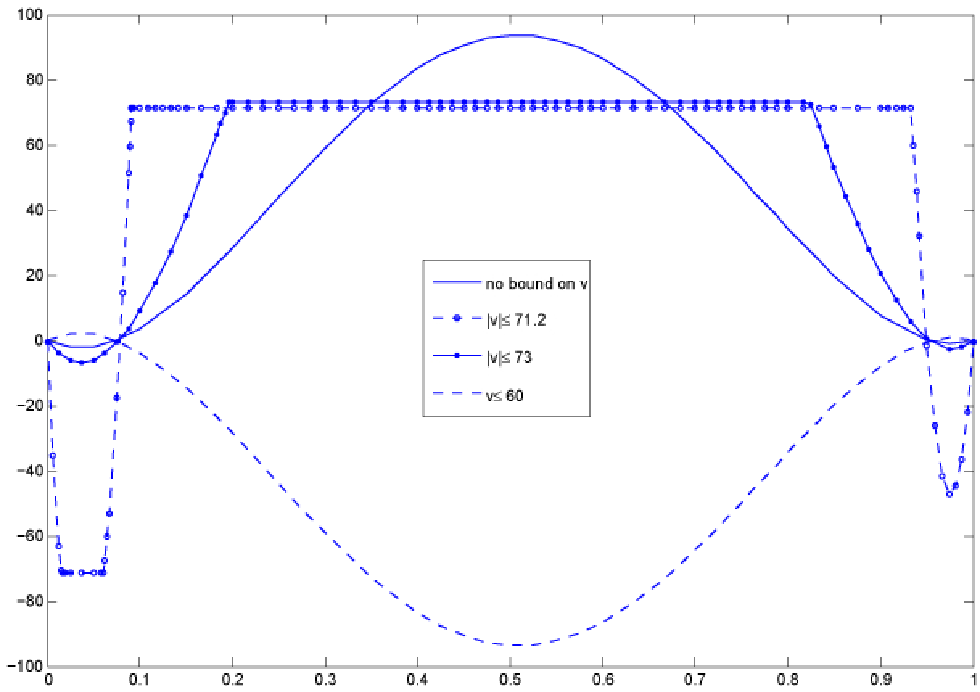

Figure 4.16: v with hard bound on auxiliary signal v, $g = 0.05$

2. Mixed bound:

$$\text{minimize } \delta^2(v) = \int_0^1 \frac{c_s}{(r^2 - v^2)^2} + c_v v^2 \, dt$$

where c_s and c_v are positive weight coefficients and r is the bound (or could be trajectory) we would like not to get close to. The reason why we have $r^2 - v^2$ instead of just $r - v$ is due to the fact if v is optimal proper auxiliary signal, than so is $-v$. Therefore we want to assure that v is not close to $\pm r$. The other power of 2 is to assure that the integrand is monotonically increasing function.

The calculated auxiliary signals for the additive uncertainty case, $g = 0$ are in figure 4.17, while results for the model uncertainty case with coefficient $g = 0.05$ are in figure 4.18. For both cases, we still have the weight on initial value of states being $P = 0$. For $g = 0$, cost parameters were $c_s = 1$, $r = 60$ and $c_v = 10^{-6}$ while for $g = 0.05$, we used $c_s = 1$, $r = 73$ and $c_v = 10^{-8}$.

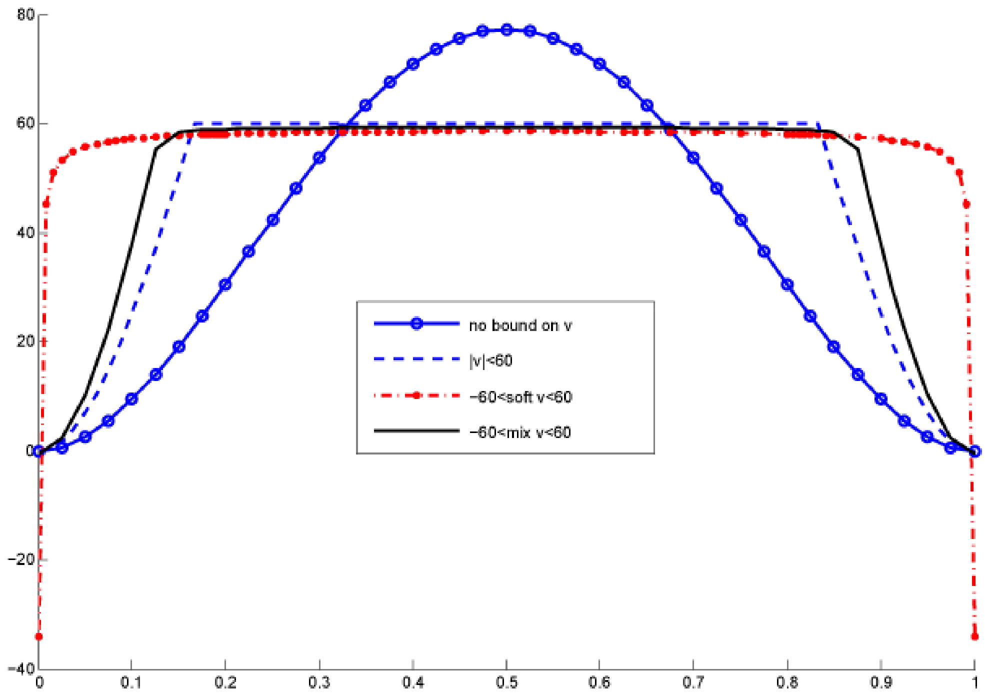

Figure 4.17: v with soft and hard bounds on auxiliary signal v, $g = 0$

The last thing we would like to point out is that if we want our auxiliary signal v to be continuous, we have to include hard bound $v_{min} \leq v \leq v_{max}$ even when we just want to use any of the mentioned soft bounds. Since SOCS is discretizing all of the data, without hard bounds we might end up with the solution as in figure 4.19 obtained for mixed cost 2 case. So for results represented in figure 4.17 we used $v_{max} = -v_{min} = 59.99$ and for figure 4.18 we used $v_{max} = -v_{min} = 72.99$ as hard bounds.

SOCS uses a continuous control approximation. It often approximates a piecewise smooth control by placing several grid points near discontinuity. Note that the grid points are closer in figure 4.19 near discontinuity. It may also have some chatter as around $t = 0.7$ in figure 4.19.

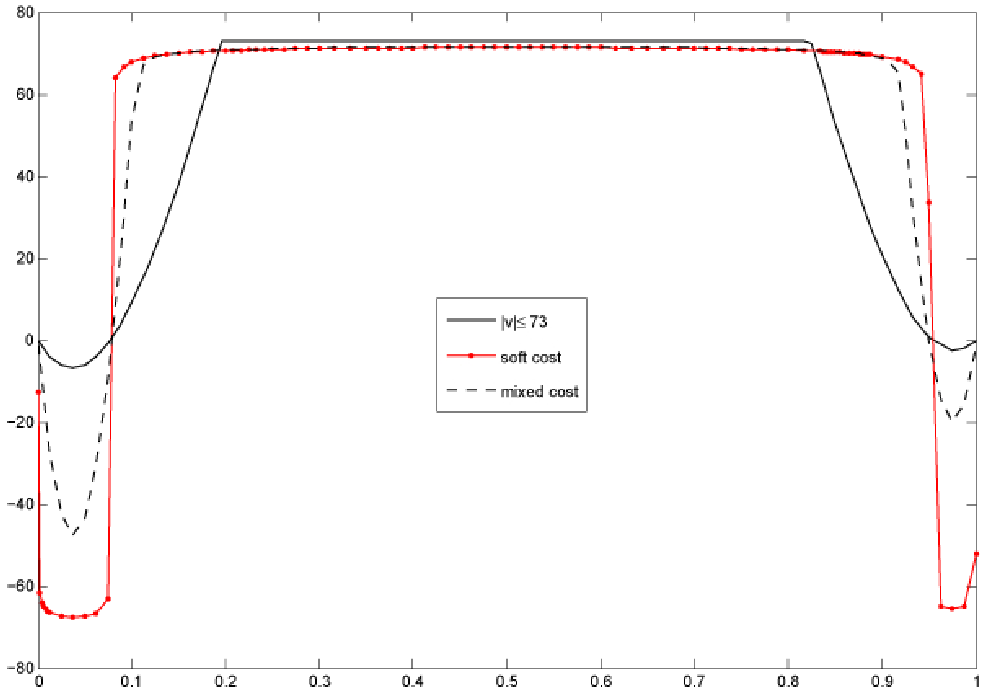

Figure 4.18: v with soft and hard bounds on auxiliary signal v, $g = 0.05$

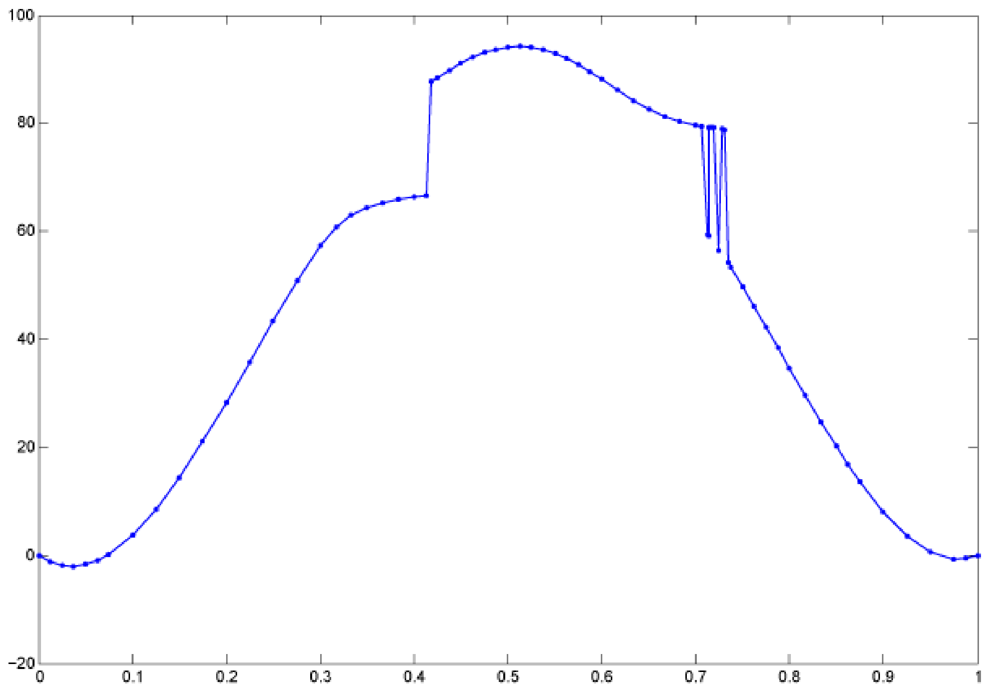

Figure 4.19: Noncontinuous v if additional hard bounds are not used

4.5.3 Bound on the States

Beside control signals, sometimes we are limited on how big our states can be during the test. Representing the problem as a big optimization problem, allows us to include such constraints. This problem is much more difficult for analysis than when we have bounds on control v, since states are part of the inner optimization problem and their derivative is also important. One way to include such constraints is within the inner optimization problem through another algebraic constraint that changes the Hamiltonian (Section 2.2). This approach would increase the complexity of the inner optimization problem. Here we present a result that is putting constraints outside of the inner optimization problem.

Here we will focus on additive noise only, $g = 0$ and constraints of the type $|x_i| \leq B_x$, where B_x is the desired bound and x_0 and x_1 are states. By observing the values of x_0 and x_1 from the case without states constraints, we chose values $B_x = 20, 18, 16$ for our bound. Proper signals v are presented in figure 4.20 while states are presented in figure 4.21. Notice that for $B_x = 16$, for some time intervals, the auxiliary signal v is not smooth at all and exhibits random behavior. The same, but not as obvious, is present for $B_x = 18$. Part of the reason for that random behavior can be explained if we observe state x_0. We can notice that during the time periods when v behaves erratically, state x_0 has exactly the value of the bound B_x. Therefore, to keep x_0 constant on the bound for that period of time, according to differential equations (4.22a), signal v has to neutralize noise component ν_0. Since noise component is random, this explains the behavior of v.

Another factor in the random appearing behavior is that if the state constraint is active, we actually have a DAE which is sometimes higher index. The direct transcription solution of optimal control problems with inequality constraints is discussed in [10],[4],[38]. The careful discussion of this topic is beyond the scope of this thesis. Suffice it to say, the result of the constraint activation can be chatter near the constraint and where the constraints change. The chatter has a regularizing affect on

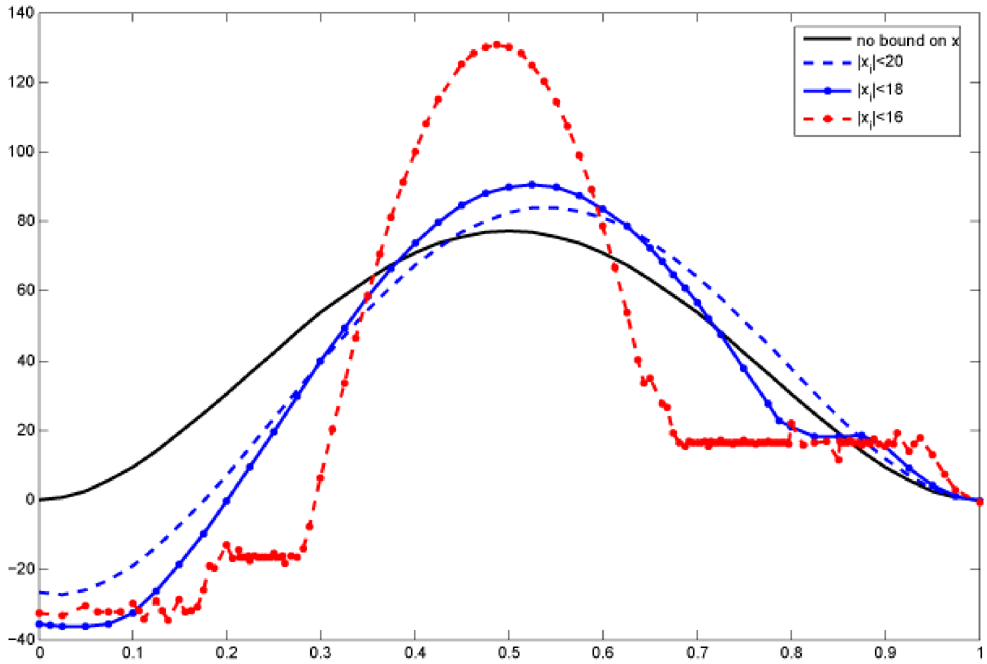

Figure 4.20: v comparison for bounds on states x_i

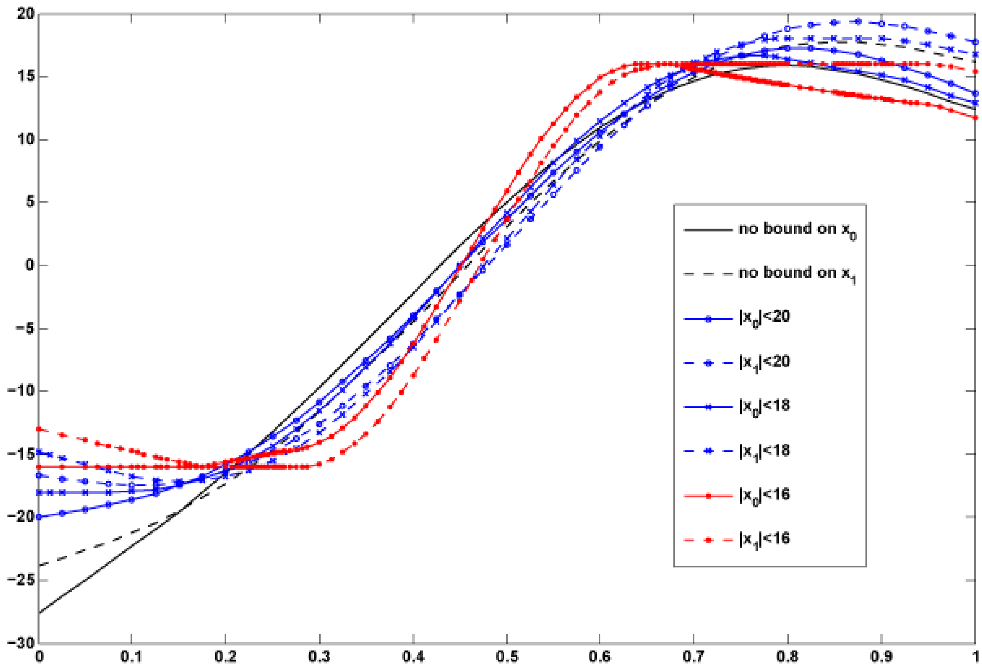

Figure 4.21: x_i comparison for bounds on states x_i

the optimizer and tends to get smaller if finer meshes are used.

Chapter 5

Summary of Contributions

Until now, the focus of active fault detection was on linear systems with additive and model uncertainty. While some special cases of nonlinear problems were mentioned [25] and analysis of when will linearization process be appropriate is done [47], this thesis is offering the first examination of trying to directly solve nonlinear problem.

One of the issues was related with the way that breaking of the noise bound was quantified (2.19d). The previously developed theory and approaches were requiring that inf and max can exchange order. While it was shown that it works for linear problems, that is not true for nonlinear problems in general. I proposed the way around it by replacing the max norm by the p-norm , and did a brief analysis of its effect in Section 3.1.1. Then I derived systems of equations representing necessary conditions and coded them into the optimization solver SOCS and used it to examine various linear and nonlinear problems. The results obtained using those codes were parts of several papers:

- I. Andjelkovic, K. Sweetingham and S. L. Campbell, *Active fault detection in nonlinear systems using auxiliary signals*, American Control Conference (ACC), Seattle, WA, 2008, to appear

- S. L. Campbell, K. J. Drake, I. Andjelkovic, K. Sweetingham, and D. Choe, *Model based failure detection using test signals from linearizations: A case study*, Proc IEEE Int. Conf Control Applications, 2006.

- K. J. Drake, S. L. Campbell, I. Andjelkovic, and K. Sweetingham, *Model based failure detection on nonlinear systems: theory and transition*, Proc. ASNE Day 2007, Arlington, VA, 2007

- K. J. Drake, S. L. Campbell, I. Andjelkovic, and K. Sweetingham, *Active incipient failure detection: A nonlinear case study*, Proc 4th International Conference on Computing, Communications and Control Technologies (CCCT), Orlando, Fl, 2006

Since the p-norm approach provides us with suboptimal proper auxiliary signals v, it was necessary to develop another SOCS driver that will evaluate the minimum amount of noise that the system requires to break a noise bound for a given v. This was done with an idea and help from Dr. Campbell. It was used as a part of the evaluation process for some papers that used the linearization approach.

- K. J. Drake, S. L. Campbell, I. V. Andjelkovic, B. L. Hannas and K. A. Sweetingham, *Applications of robust failure detection algorithms to power systems*, Proc. 13th Intelligent Systems Application to Power Systems (ISAP),Washington, DC, 2005

- S. L. Campbell, K. J. Drake, I. Andjelkovic, K. Sweetingham, and D. Choe, *Model based failure detection using test signals from linearizations: A case study*, Proc IEEE Int. Conf Control Applications, 2006.

Using developed codes for nonlinear cases, in this thesis the examples and analysis of nonlinear problems where these codes are valuable asset are presented in Chapter 4. The examples with additive noise range from the problems where the linearization can not provide any result to the problems where linearization and nonlinear p-norm approach produced quite similar auxiliary signals v. Since nonlinear problems are notorious for their potential of having multiple local minimums and maximums, and the importance of a good initial guess for the nonlinear solver like SOCS, special attention was given to those issues. There is a potential that the p-norm approach can be an alternative even for linear problems. The expectation is due to the fact that p-norm does not require an additional parameter β that is necessary for traditional approach. In a few examples that were done, we were not able to confirm that premise. We did observe though, that we can make the p-norm solution approach the optimal solution for linear problems by increasing the value of p. We also dedicated special attention to the different types of cost functions and how they affect our solution.

We also derived a system of equations and implemented them in SOCS that represent linear problems with model uncertainty as a big optimization problem. This was the first time this was done. For this type of problem, the approach using Riccati equations programmed in Scilab exists [8], however having a big optimization problem formulation allows us to tackle some additional cases, such as when there are constraints on v and states x_i. We illustrated that with several examples. This was the first solution of a constrained direct optimization formulation of a model uncertainty problem. We also presented a few examples where SOCS slightly outperformed the Scilab solution by having a smaller cost v. We also pointed out the issue of local minimums that can occur.

During my studies, one of the most valuable practical experiences I had, was related with the project at NAVSEA, Philadelphia, with Kimberly J. Drake as a project leader. We were observing a system consisting of an air pump, tank and connecting pipes, and we were trying to be able to detect potential leakage of the tank by using an active fault detection approach. This is the first real life case where we are trying to use active fault detection principles. A good model is crucial for active fault detection, and I was a part of the team that was trying to determine the best way to model this physical system. The research is ongoing, and produced the following previously mentioned papers so far:

- K. J. Drake, S. L. Campbell, I. Andjelkovic, and K. Sweetingham, *Model based failure detection on nonlinear systems: theory and transition*, Proc. ASNE Day 2007, Arlington, VA, 2007

- K. J. Drake, S. L. Campbell, I. Andjelkovic, and K. Sweetingham, *Active incipient failure detection: A nonlinear case study*, Proc 4th International Conference on Computing, Communications and Control Technologies (CCCT),

As a part of the side project, by the idea and algorithm from Dr. Campbell, I wrote a driver that investigated potentials of using SOCS for the delay problems with minimal SOCS modifications. This is an area of future research and the preliminary results were mentioned in

- Stephen L. Campbell, J. T. Betts, Anna Engelsone, and Ivan Andjelkovic, *Whats new with direct transcription methods for optimal control problems*, talk-presentation at Boeing, November 17,2005

Chapter 6

Future Work

In this chapter we will present some areas for future research and mention some possible extensions of the results obtained in this thesis.

6.1 p-norm Approach for Model Uncertainty

One idea for future research is to try to implement the p-norm approach for model uncertainty in a slightly different way. In [7], the approach to solving and analyzing model uncertainty was to represent it in the form similar to additive uncertainty. We had that for additive uncertainty case, for the noise bound to be broken,

$$S = (x(0) - x_0)^T P^{-1}(x(0) - x_0) + \int_0^T \nu^T I \nu \; dt \geq 1 \tag{6.1}$$

had to be valid. At the same time, for the model uncertainty case only

$$S = (x(0) - x_0)^T P^{-1}(x(0) - x_0) + \int_0^s \mu^T I \mu - \xi^T I \xi \; dt \geq 0 \tag{6.2}$$

had to be true for some $s \in [0, T]$. Note, that the model uncertainty case defined as above already introduces more conservatism, i.e. allows more noise than necessary. It still guarantees proper auxiliary signal v, but it will not be optimal.

For cases where both uncertainties are present, the new noise bound evaluation was achieved by putting additive and model uncertainties together in the form

$$S = (x(0) - x_0)^T P^{-1}(x(0) - x_0) + \int_0^s \varphi^T \Gamma \varphi \; dt \geq 1 \tag{6.3}$$

with $\varphi = \begin{bmatrix} \nu \\ \mu \\ \xi \end{bmatrix}$ and $\Gamma = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & -I \end{bmatrix}$.

This presentation is introducing even more conservatism in our problem by letting additive noise component ν potentially mask breaking of the noise bound by model uncertainty constraint. This type of formulation was necessary in [7] to get a single noise bound required by Riccati approach. But in principle a direct optimization could handle several noise bounds at once, and we might try to exploit that.

In this thesis, when dealing with additive uncertainty only, in the p-norm approach we introduced variable z that is in effect measuring amount of additive noise in the system. Since we were distinguishing between 2 models, we had 2 such variables.

Let's introduce a total of 4 such variables, 2 for each model. One will still be measuring additive noise in the system as was done in this thesis

$$\dot{z_a} = \|\nu\|_2^2 \,, z_a(0) = (x(0) - x_0)^T P^{-1}(x(0) - x_0) \tag{6.4}$$

while the other will measure model uncertainty

$$\dot{z_m} = \|\mu\|_2^2 - \|\xi\|_2^2 \,, z_m(0) = 1 \tag{6.5}$$

Notice that $z_m(0) = 1$. The reason is requiring both variables to break the noise bound at the same value, in this case 1. So, breaking of the noise bound when we observe 2 models will be equivalent to

$$\max\{z_{a0}(T), z_{a1}(T), z_{m0}(s_0), z_{m1}(s_1)\} \geq 1 \tag{6.6}$$

If we can assure that all of the above variables are non-negative, we can replace max norm in the above constraint by p-norm .

$$\inf_{x_i,\nu_i,y,u,s\in[0,T]} z_{a0}(T)^p + z_{a1}(T)^p + z_{m0}(s_0)^p + z_{m1}(s_1)^p \geq 4 \tag{6.7}$$

There are several issues to be resolved:

1. **p-norm works only if we can assure that variable is positive**

   That is not an issue for z_{ai} since it's derivative is always positive. However, that is not the general case for z_{mi}. To avoid this issue, we can either require the limit on the model uncertainty variable $\xi < \frac{1}{T}$, where T is time interval, or we might shift noise bound of 1 to some higher value.

2. **How can SOCS handle different times s_0, s_1 and T**

   One way is to solve system by multiple phases. First phase has dynamics as determined by the necessary conditions. If s_i is reached before time T, SOCS goes into the second phase. The constraint equations of the second phase are the same as in the first one, with the exception that $\dot{z_{mi}} = 0$. Then we can just evaluate all our z variables at the time T to establish whether the bound is broken.

$$\inf_{x_i,\nu_i,y,u,s\in[0,T]} z_{a0}(T)^p + z_{a1}(T)^p + z_{m0}(T)^p + z_{m1}(T)^p \geq 4 \tag{6.8}$$

Since the p-norm is introducing conservatism as well, it will be interesting to find out which approach is producing better non-optimal auxiliary signals v (better in terms of the definition of cost, L^2 norm or any other way we decide to evaluate v).

6.2 Delay Implementation in SOCS

All the research done in this thesis was neglecting the effect of delays in the systems. It is a fact of life that delay often exists, whether it is due to the signal processing, reaction time, signal propagation or any other reason. While in a lot of instances the effect of delays can be ignored, there are numerous examples where that is not the case. Detailed research of the effects of delay on active fault detection was done [18], [12], [13], [14], [29], [30]. Recently [9], another novel idea by Dr. Campbell was introduced. SOCS does not have built in a mechanism to deal with delays. The idea was the way how to implement delays in SOCS with the minimal change of the program itself.

The approach is based on dynamic iterations. If our observed delayed differential equation

$$\dot{x} = f(x(t), x(t - \tau(t)), u(t), t) \tag{6.9}$$

is replaced with discretized version

$$\dot{x_n}(t) = f(x_n(t), x_{n-1}(t - \tau(t)), u(t), t) \tag{6.10}$$

the theory states [48],[49] that $lim_{n \to \infty} x_n = x$. The idea is to utilize dynamic iterations by defining outside loop for SOCS. Instead of solving original problem (6.10), SOCS is solving

$$\dot{x_n}(t) = f(x_n(t), z_{n-1}, u(t), t) \tag{6.11}$$

Where z_{n-1} is a fixed function for a given iteration, obtained during the previous iteration as a delayed version of x_{n-1}. The algorithm is presented in figure 6.1.

As a part of my research, I ran a few initial samples with constant delay and the results were promising. The examples I ran were relatively simple linear problems. However, there is a lot of theoretical work and analysis that has to be done. Also, extensions where we will allow delay to vary with time still have to be implemented. Applications on more complex problems might reveal additional issues that will have to be resolved. It would be beneficial if combined with p-norm approach, delays could be implemented for nonlinear problems as well.

6.3 Other Extensions

During the PhD research, the author encountered several topics that he would like to explore in more depth.

1. How to choose an initial guess for our solver

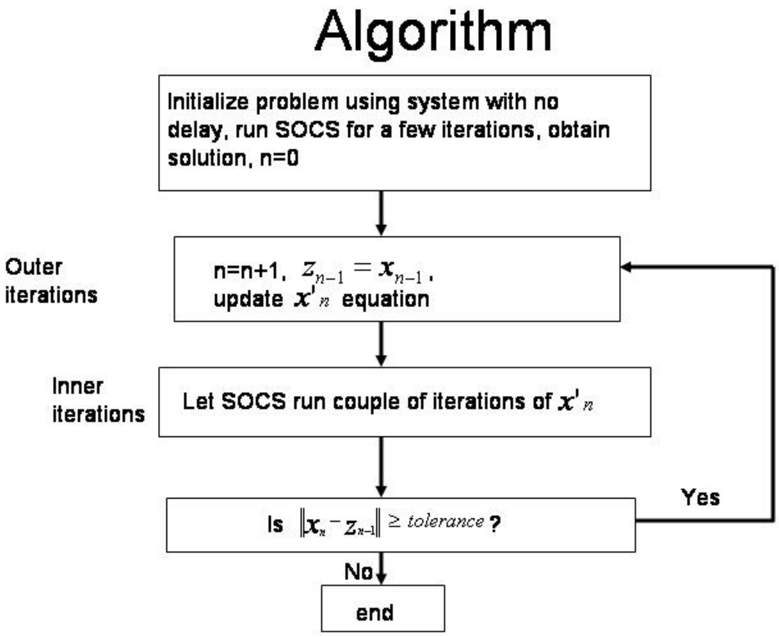

Figure 6.1: SOCS implementation of problems with delay

2. Investigate why in Example 4.1.3 the simplified problem gave a good initial guess

3. Implementing the better interpolation method for noise bound finder

4. Impact of discretization-integration method of choice

5. Investigating impact of using second order term from Taylor expansion

Topic 1 and the related more specific topic 2 are dealing with initial guesses. As noted during the thesis, in many examples, a good initial guess makes the difference in whether we will get stuck in a local minimum, or whether SOCS would be able to get any solution at all. Beside random guesses, we were using solutions of simplified systems as our initial guess. Namely, we were using the solution of a linearized problem as the initial guess of a nonlinear problem, and in one occasion in Example 4.1.3 we ignored one noise component to obtain an initial guess. While those approaches worked in the observed examples, there is still a lot of theoretical work to be done to be able to quantify and predict success of chosen initial guess approach.

Topics 3 and 4 are related with the area of numerical analysis and SOCS implementation. The purpose of topic 3 is to improve our SOCS driver described in the Section 3.4. While this driver is currently used to estimate the noise bound, the approach can be used in the problems with delay mentioned in Section 6.2. Regarding topic 4, sometimes just changing of integration-discretization method made the difference whether results is obtained or not. Investigating this issue in more depth would be interesting.

Finally, issue 5 is an extension of the standard linearization approach. For practical purposes, we usually approximate nonlinearities by using only up to linear term in Taylor series in the process of linearization. It is obvious that improvement of the approximation, at the expense of complexity, could be achieved by using quadratic approximation where we will use quadratic term of the Taylor series as well.

Bibliography

[1] I. Andjelkovic, K. Sweetingham and S. L. Campbell, *Active fault detection in nonlinear systems using auxiliary signals*, American Control Conference (ACC), Seattle, WA, 2008, to appear

[2] M. Athans and P. Falb, *Optimal control*, McGraw-Hill, New York, 1966

[3] J. T. Betts and W. P. Huffman, *Sparse optimal control software socs*, Mathematics and engineering technical document MEA-LR-085, Boeing information and support services, 1997

[4] J. Betts, N. Biehn, and S. L. Campbell, *Convergence of nonconvergent IRK discretizations of optimal control problems with state inequality constraints*, SIAM J. Sci. Comp.,23,pg:1981-2007, 2002

[5] M. Blanke, R. Izadi-Zamenabadi, S. Bogh and C Lunan, *Fault-tolerant control systems*, Control engineering practice -CEP, 5(5), pg.693-702, 1997

[6] *SOCS manual*, release 6.4.9, Boeing company, 2007

[7] S. L. Campbell, and R. Nikoukhah, *Auxiliary signal design for failure detection*, 1st ed. Princeton, New Jersey, 2004., ISBN 0-691-09987-1

[8] Stephen L. Campbell, Jean-Philipe Chancelier, and Ramine Nikoukhah, *Modeling and simulation in Scilab/Scicos*, Springer, 2005

[9] Stephen L. Campbell, J. T. Betts, Anna Engelsone, and Ivan Andjelkovic, *Whats new with direct transcription methods for optimal control problems*, talk-presentation at Boeing, November 17,2005

[10] S. Campbell, N. Biehn, L. Jay, and T. Westbrook, *Some comments on DAE theory for IRK methods and trajectory optimization*, J. Comp. Applied Math., 120, pg:109-131, 2000

[11] S. L. Campbell, K. J. Drake, I. Andjelkovic, K. Sweetingham, and D. Choe, *Model based failure detection using test signals from linearizations: A case study*, Proc IEEE Int. Conf Control Applications, 2006.

[12] S. Campbell, K. Drake, and R. Nikoukhah, *Auxiliary system design for multi-model identification in systems with multiple delays*, 10th Mediterranean Conference on Control and Automation, Lisbon, July 7-11 2002.

[13] S. Campbell, K. Drake, and R. Nikoukhah, *Analysis of spline based auxiliary signal design for failure detection in delay systems*, Proc. IEEE CSMC, Washington DC, October 2003

[14] S. Campbell, K. Drake, R. Nikoukhah, and F. Delebecque, *Rapid multi-model identification in systems with delays*, 3rd IFAC Workshop on Time Delay Systems (TDS 2001),pg.296-301, Sante Fe, NM, Dec. 8-10 2001

[15] Benoît C. Chachuat, *Nonlinear and dynamic optimization: from theory to practice*, Laboratoire d'Automatique, École Polytechnique Fédérale de Lausanne, Lectures for Winter Semester 2006/2007

[16] D. Choe, S. L. Campbell and R. Nikoukhah, *A comparison of optimal and suboptimal auxiliary signal design approaches for robust failure detection*, IEEE Conference on Control Applications, Toronto, 2005

[17] D. Choe, *Digital signal design for fault detection in linear continuous dynamical systems*, PhD thesis, North Carolina State University, Raleigh, 2007

[18] K. Drake, *Analysis of numerical methods for fault detection and model identification in linear systems with delays*, PhD thesis, North Carolina State University, 2003

[19] K. J. Drake, S. L. Campbell, I. Andjelkovic, and K. Sweetingham, *Model based failure detection on nonlinear systems: theory and transition*, Proc. ASNE Day 2007, Arlington, VA, 2007

[20] K. J. Drake, S. L. Campbell, I. Andjelkovic, and K. Sweetingham, *Active incipient failure detection: A nonlinear case study*, Proc 4th International Conference on Computing, Communications and Control Technologies (CCCT), Orlando, Fl, 2006

[21] K. J. Drake, S. L. Campbell, I. V. Andjelkovic, B. L. Hannas and K. A. Sweetingham, *Applications of robust failure detection algorithms to power systems*, Proc. 13th Intelligent Systems Application to Power Systems (ISAP),Washington, DC, 2005

[22] A. Engelsonne, *Direct transcription methods in optimal control: theory and practice*, PhD thesis, North Carolina State University, Raleigh, 2006

[23] Christodoulos A. Floudas et.all, *Handbook of test problems in local and global optimization*, Kluwer, Netherlands, 1999

[24] H. H. Goldstine, *A history of the calculus of variations from the 17th to the 19th Century*, Springer, New York,1981

[25] K. Horton, *Fault detection and model identification in linear dynamical systems*, PhD thesis, North Carolina State University, Raleigh, 2001

[26] David G. Hull, *Optimal control theory for applications*, Springer, NY, 2003

[27] R. Isermann, *Fault-diagnosis systems*, Springer, Heidelberg, 2006, ISBN-10 3-540-24112-4

[28] R. Isermann, *Fault diagnosis of technical processes - applications*, Springer, Heidelberg, 2006

[29] Kazafumi Ito, and Franz Kappel, *Evolution equations and approximations*, vol.61 of series on advances in mathematics for applied sciences, World Scientific Pub, River Edge, NJ, 2002

[30] Kazafumi Ito, and Franz Kappel, *A uniformly differentiable approximation scheme for delay systems using splines*, Applied Math. Optimization 23, pg.217-262, 1991

[31] Johannes Jahn, *Introduction to the theory of nonlinear optimization*, Springer, NY, 2007

[32] R. Baker Kearfott, *Rigorous global search: Continuous problems*, Kluwer, Netherlands,1996

[33] D. E. Kirk, *Optimal control theory*, Englewood Cliffs, Prentice-Hall, New Jersey, 1970

[34] P. Kunkel and V. Mehrmann, *Differential-algebraic equations*, 1st ed. European Mathematical Society, Zürich, 2006., ISBN 3-03719-017-5

[35] F. L. Lewis, and V. L. Syrmos. *Optimal control*, 2nd ed. John Wiley and Sons, inc, New York, 1995., ISBN 0-471-03378-2

[36] Helmut Maurer, and Hans Joachim Oberle, *Second order sufficient conditions for optimal control problems with free final time: the Riccati approach*,SIAM J. Control Optimization (2002), vol 41. pg.380-403

[37] K. M. Moudgalya, and V. Ryali, *A class of discontinuous dynamical system I. An ideal gas-liquid system*, Chemical engineering sciences, Vol.56, pg. 3595-3609, 2001

[38] R. Nikoukhah, S. L. Campbell, and F. Delebecque, *Auxiliary signal design for failure detection in uncertain systems*, Proc. IEEE Med. Conf. Control and Automation, 2001

[39] R. Patton, P. Frank, and P. Clark, *Issues of fault diagnosis for dynamic systems*, Springer, New York, 2000

[40] János D. Pintér, *Global optimization: software, test problems, and applications*, Handbook of global optimization volume 2, Kluwer Academic Publishers, Netherlands, 2002

[41] M. Polycarpou, A. Vemure and A. Ciric, *Nonlinear fault diagnosis of differential algebraic systems*, Proc IFAC Symposium on Fault Detection, Supervision and Safety for Technical Process, vol.1, pg.510-515, August 1997

[42] L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze and E. F. Mischenkko, *The mathematical theory of optimal processes*, Wiley, New York, 1962

[43] R. W. H. Sargent, *Optimal control*, Journal of Computational and Applied Mathematics 124 (2000) 361-371

[44] A. Seierstad and K. Sydsaeter, *Sufficient conditions in optimal control theory*, International Economic Review, 18(2):367-391, 1977

[45] Roman G. Strongin and Yaroslav D. Sergeyev, *Global optimization with non-convex constraints*, Kluwer, Netherlands, 2000

[46] H. J. Sussmann and J. C. Willems, *300 years of optimal control: from the brachystochrone to the maximum principle*, IEEE Control Systems (1997) 32-44

[47] K. Sweetingham, *Auxiliary signal design for fault detection in nonlinear systems*, PhD thesis, North Carolina State University, Raleigh, 2008

[48] S. Vandewalle, *Dynamic iteration methods for delay partial differential equations*, 9th seminar of numerical solutions of differential and differential algebraic equations, Halle, Germany, September 4-8, 2000

[49] S. Vandewalle, *Waveform relaxation for delay partial equations*, International conference on scientific computation and differential equations, SCICADE99, Frase Island, Queensland, Australia, August 13, 1999

[50] J. S. Yang, *Mixed H^2 compensator design for an aircraft control problem*, Proceedings of the IEEE conference on decision and control, pg.1964-1969, Phoenix, AZ, 1999