CONTROLLER DESIGN OF DECENTRALIZED SINGULARLY PERTURBED SYSTEMS

A Thesis by

Satya Vineela Danthuluri

Bachelor of Engineering, Andhra University, 2008

Submitted to the Department of Electrical Engineering and Computer Science
and the faculty of the Graduate School of
Wichita State University
in partial fulfillment of
the requirements for the degree of
Master of Science

December 2010

CONTROLLER DESIGN OF DECENTRALIZED SINGULARLY PERTURBED SYSTEMS

The following faculty members have examined the final copy of this thesis for form and content, and recommend that it be accepted in partial fulfillment of the requirement for the degree of Master of Science with a major in Electrical Engineering

____
John M. Watkins, Committee Chair

____
Edwin M. Sawan, Committee Member, Advisor

____
Brian Driessen, Committee Member

DEDICATION

To my parents and my sister—Varma, Geetha, and Vinuthna—
and all the teachers who provided me their valuable guidance

ACKNOWLEDGEMENTS

I extend my heartfelt gratitude to my academic advisor, Dr. Edwin Sawan, for his endless patience and brilliant guidance throughout my graduate studies. His constructive criticism and invaluable support during every stage of my master's thesis helped me to mould myself better in many aspects. I thank him very much for being such a great inspiration.

I am very grateful to Dr. John Watkins, my committee chair, for the support he extended and his interest in me. The courses he taught and his suggestions were of tremendous assistance in the completion of my thesis work. I cannot thank him enough for being my committee chair, despite his busy schedule.

I also express my sincere appreciation to Dr. Brain Driessen, from the Department of Mechanical Engineering, for serving as a member of my thesis committee.

I am extremely thankful to all the teachers in my life, from kindergarten through my graduate studies, for helping me become the person I am now and for being a constant source of support during every stage of my life.

ABSTRACT

In this thesis, the performance of a decentralized system was analyzed based upon the design of the controller, either a reduced-order model or a full-order model. The singular perturbation technique was used to obtain the reduced-order model of a decentralized system. Then this model was used to design a controller for state feedback. Since this controller was the reduced-order model of the system, it was implemented based on the full-order model by padding zeroes. Thus, the controller was an approximation. Hence, a performance analysis was conducted to verify the near approximation of the design using the singular perturbation technique. To check the sensitivity of the design, the performance of the system using the controller designed on the full-order system was compared with the performance of the system using the controller designed on the reduced-order model. A comparison of the results showed that the effectiveness of the reduced-order model design can be verified and checked as to whether it is giving the required approximate results for a lower-cost controller design.

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Motivation and Objective

Currently, most systems in practical scenarios are both complex in nature and large in terms of their parameters. These are referred to as large-scale systems. Several methods are used to analyze these systems. One method is to consider the entire system as an interconnection of subsystems, which allows for simplified computations. Such a system, which is divided into individual subsystems, is known as a decentralized system. Controlling the decentralized system is much simpler because of each subsystem’s local controller and is economical because it does not involve tedious data gathering, storage requirements, and computational procedures. Although the full-order decentralized control is easy to compute, sometimes it is costly if the system has a very large number of subsystems, which in turn need controllers. Hence, if the system is having slow- and fast-response transients, a reduced-order model can be obtained using the singular perturbation technique. Thus, the cost of controller will be reduced at the price of some accuracy. The accuracy of the system control depends on the value of epsilon (ε), the parameter of the singular perturbation technique.

The main objective of this thesis was to compare the performance analysis of a system when the controller was designed using both a full-order model and a reduced-order model. Based on the results comparison, the accuracy of the reduced-order model design was verified with that of decentralized control.

1.2 Overview

In this thesis, controller design and performance analysis of a decentralized system is presented. Chapter 2 provides a review of the theoretical background for a decentralized control system, the singular perturbation technique, and the decentralized singularly perturbed model. In Chapter 3, design of a decentralized control for a full-order system is given along with a numerical example. Chapter 4 provides the procedure for controller design using a singular perturbation technique for both state-feedback and output-feedback systems. An example is provided to explain the procedure numerically. In Chapter 5, a decentralized singularly perturbed system is considered, and a controller is designed using the two above-mentioned methods. Comparison of performance is done with an illustrated example. Chapter 6 presents the conclusions and suggestions for future work.

CHAPTER 2

THEORETICAL BACKGROUND

2.1 Introduction

In this chapter, some essential theoretical concepts that will be used in the following chapters are presented. These methods are helpful in designing a controller for a full-order system. First, a decentralized system is explained and concepts of the decentralized control system are given. Second, the singular perturbation technique is explained. This method was used to obtain a reduced-order model from a full-order model. Feedback control for a reduced-order model is discussed, as well as how to implement it on a full-order model. Third, the details of a decentralized singularly perturbed model are introduced. Based on these basic ideas, the thesis work will be presented in later chapters.

2.2 Decentralized Control System

A large-scale system that can be controlled by several small-scale systems located either nearby or far away is known as a decentralized control system. The method of partitioning the full-scale system for the purpose of designing a control for it is known as decentralization. In the decentralized system, each subsystem plays an important role in controlling the full-scale system. The basic idea here is to design a local controller that accompanies each individual subsystem. This local controller is concerned only with the output of the local subsystem, not the output related to other subsystems. In turn, this results in fewer storage requirements and simpler computational procedures. Figure 1 shows a block diagram of a decentralized system with n individual subsystems.

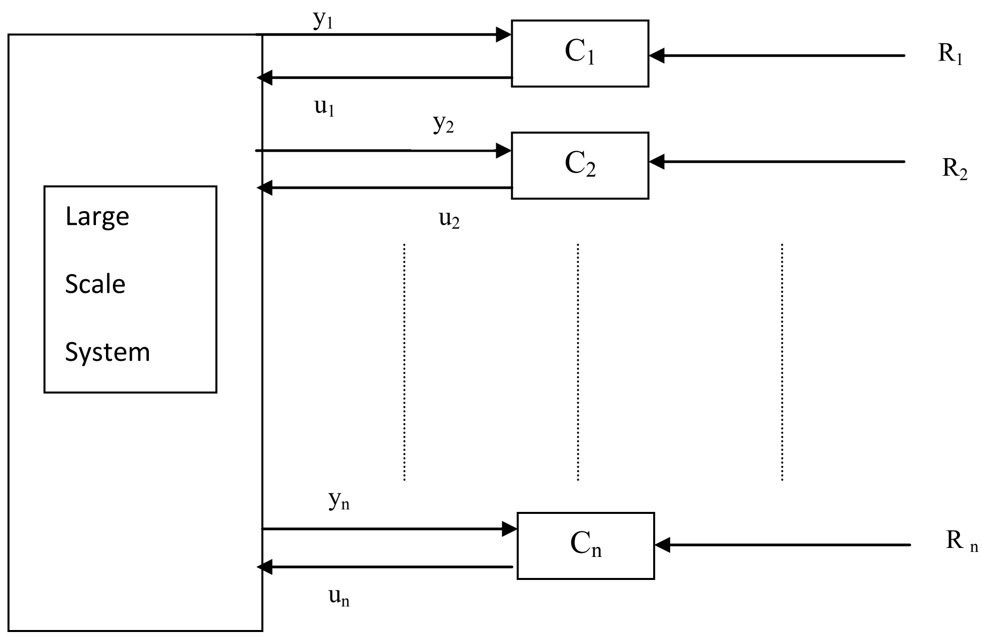

Figure 1: Decentralized system having n subsystems.

The marked characteristic of a decentralized system is that individual subsystems or their local controllers do not intercommunicate. In other words, the control of a given subsystem is dependent only on the external input and output of that system itself, and is independent of inputs and outputs from other subsystems. In the decentralized system shown in Figure 1, there are n controllers, namely C_1, $C_2 \ldots C_n$, u_1. The inputs are $u_2, \ldots u_n$, the outputs for the large-scale system are y_1, $y_2, \ldots y_n$, and the reference inputs to the controllers are R_1, $R_2, \ldots R_n$. The control u_1 from controller C_1 is determined based on y_1 and R_1. Similarly, the control u_2 is determined depending on y_2 and R_2. In general, the design of all these controllers, C_1, $C_2 \ldots\ldots$, C_n, can be considered as independent output feedback-control problems of subsystems that have smaller orders than that of the full system. Hence, this procedure concentrates on dividing a full-scale system into subsystems, designing a controller for each subsystem, and implementing it on a full-scale system. Each subsystem control can be implemented on a full-scale system, once the gains of each local controller are determined. Let K_1, $K_2, \ldots$, K_n be the gains of the local controllers.

Then the controller for a full-scale system is given as a block diagonal matrix with individual gain matrices as diagonal elements. The controller of decentralized system, K, is given as

$$K = \begin{pmatrix} K_1 & & \cdots & 0 \\ \vdots & K_2 & & \vdots \\ & & \ddots & \\ 0 & \cdots & & K_n \end{pmatrix}$$

In this way, the control of a large-scale system is obtained using reduced calculations. Also, the cost of control will be less, as can be seen from matrix K, where only diagonal elements and all off-diagonal elements are zero. Thus, a decentralized control structure has been constructed.

2.3 Singular Perturbation Method

2.3.1 Introduction

In the singular perturbation method, a reduced-order model of a full-order system is used to design the controller of the system. This reduced-order model is obtained from a singularly perturbed model in which the separation of time scales is clearly distinct. In other words, a singularly perturbed model is one in which the responses of transient vectors have been clearly divided into slow-response transients and fast-response transients. Slow-response transients are those states having dominant poles, i.e., poles closer to the imaginary axis. Fast-response transients are those states having weak poles, i.e., poles farther from the imaginary axis. To obtain a simplified or reduced-order model, we consider only the state vectors having dominant behavior and thus neglect fast-response state vectors. The mathematical calculations to obtain a reduced-order model and further computations are provided in the following subsections.

2.3.2 Reduced-Order Model of a System

A reduced-order model can be obtained from a full-scale model only if the system has both quasi-steady states having slow-response vectors and some states having fast-response vectors. Let us consider a model in state variable form [3]:

$$\dot{x} = f(x, z, \varepsilon, t) \qquad x(t_o) = x^0,\ x \in R^n \tag{2.1}$$

$$\varepsilon \dot{z} = g(x, z, \varepsilon, t) \qquad z(t_o) = z^0,\ z \in R^m \tag{2.2}$$

where x and z are state vectors, t is time, and ε is a singular perturbation parameter. In this system, there is a clear distinction of states, since some states have their derivatives multiplied by a small parameter ε, which takes only small positive values. Now the reduced-order system in the new time scale should be obtained. Let T be the new time scale variable. For the above system,

$$\varepsilon\, dz/dt = dz/dT \qquad \text{and } dT/dt = 1/\varepsilon$$

Now the scalar ε becomes the parameter of a singular perturbation. Setting ε = 0 gives the reduced-order model. As ε = 0, then $\varepsilon \dot{z} = 0$. Then the differential equation (2.2) is transformed into an algebraic equation g (x, z, ε, t) = 0. Now it is necessary to find the root of this equation for z and substitute it back into equation (2.1). This process eliminates z and ε from equation (2.1), thus resulting in the system equation as

$$\dot{x}' = f'(x', t) \qquad x'(t_o) = x^0 \tag{2.3}$$

where x' is the state vector of the reduced-order model.

Equation (2.3) represents a reduced-order model of order n, whereas the full-order system is of order (n + m). This reduced-order model is used for designing the controller, which is described in later subsections of this chapter.

2.3.3 State Feedback Control

If control to the system is provided by accessing a few of the states, then multiplying them with feedback gain K_1 and giving –ve feedback is known as state feedback control, which is shown in Figure 2. Consider that K_1 is the feedback gain for the system shown here. Then the state feedback control is given by $u' = - K_1 x'$. With the controller included, the system equation for the closed-loop system can be written as $\dot{x}' = A_s x'$. Then the system matrix A_s is given as $A_s = A' - B' K_1$. The characteristic equation of the system will be given as $\Delta_a = \det(s I_{nxn} - A_s) = 0$. Let $\Delta_d = 0$ be the desired characteristic equation. To obtain K_1, the condition is $\Delta_a = \Delta_d = 0$. Thus, by solving the given condition, K_1, the state feedback gain of the controller, is obtained.

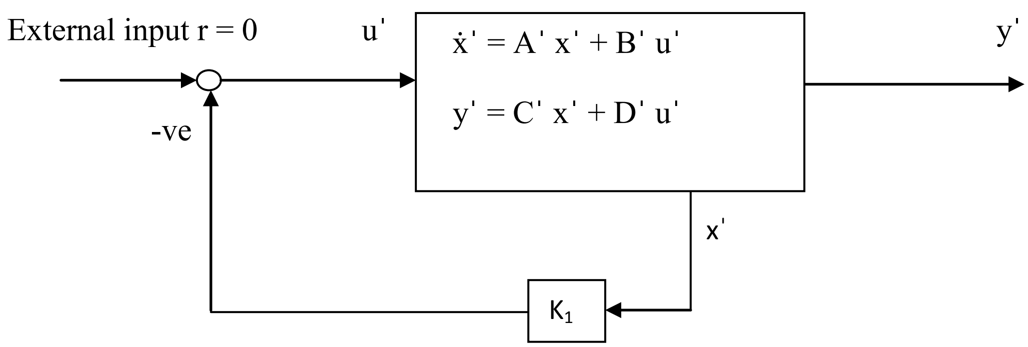

Figure 2: State feedback control

2.3.4 Output Feedback Control

If the feedback of the system is given as feedback after multiplying with the output feedback gain of the controller, then it is considered an output feedback system. Let K_2 be the gain for the output feedback controller of that system that is shown in Figure 3. The output feedback control is given by $u' = K_2 y'$. In order to obtain a closed-loop system equation, substitute the output control equation in the open-loop system equations as follows:

$$y' = C' x' + D' K_2 y'$$

$$y' = (I - D' K_2)^{-1} C' x'$$

$$u' = K_2 \, (I - D' \, K_2)^{-1} \, C' \, x'$$

$$\dot{x}' = (A' + B' \, K_2 \, (I - D' \, K_2)^{-1} \, C' \,) \, x' \tag{2.4}$$

Then the system matrix A_o is $A_o = (A' + B' K_2 (I - D' K_2)^{-1} C')$. The characteristic equation of the system is given as $\Delta_a = \det(s \, I_{nxn} - A_o) = 0$. Let $\Delta_d = 0$ be the desired characteristic equation. To obtain K_2 the condition is $\Delta_a = \Delta_d = 0$. Thus by evaluating the given condition, K_2, the output feedback gain of controller is obtained.

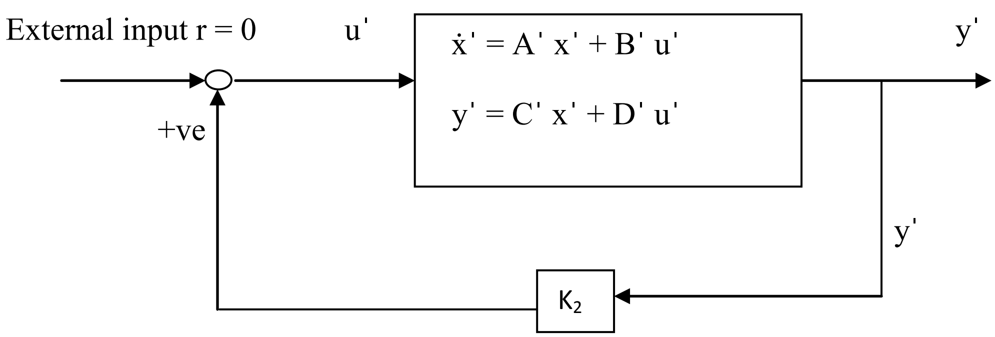

Figure 3: Output feedback control

2.3.5 Implementation to Full-Order Model

In the above two subsystems, an outline to the design controller was presented for the reduced-order model of order n. However, the main goal here is to design a controller for a full-order system of order (n + m). Hence, the obtained controller needs to be implemented on a full-order model. Depending on the order of matrices A, B, C, and D of the full-order model, the order of the gain matrix K of the controller was calculated. For example, if A is of order 4 x 4 and B is of order 4 x 2, then a controller for state feedback needs to be designed. So the order of the controller for the full-order system needs to be of order 2 x 4. Consider that out of the four state vectors of the system, two states have fast transients and two states have slow transients. Then the reduced-order model is order 2 x 2, and the resulting controller has a gain matrix K_1 of order 2 x 2. In this case, matrix K_1 is implemented to a 2 x 4 matrix by padding a sufficient number of zeroes.

$$K_1 = \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix} \qquad K = \begin{bmatrix} K_1 & 0 & 0 \\ K_2 & 0 & 0 \end{bmatrix}$$

$$K = \begin{bmatrix} K_{11} & K_{12} & 0 & 0 \\ K_{21} & K_{22} & 0 & 0 \end{bmatrix}$$

Thus, the resultant controller for the full-order system is obtained. In general, if K_r, the feedback gains matrix obtained for the reduced-order model, is of order mxm and needs to be implemented to a full-order system having controller matrix K of order nxn, where n>m, then K is given as

$$K = \begin{pmatrix} K_r & 0 & \cdots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & & \vdots \\ 0 & \cdots & & 0 \end{pmatrix} \tag{2.5}$$

2.4 Decentralized Singularly Perturbed Model

A decentralized singularly perturbed model is a decentralized system with a number of interconnected individual subsystems represented by a separation of state vectors in time scales. For this system, a few subsystems have dominant poles, and a few subsystems have weak poles. Figure 4 shows the pictorial representation of one such system.

From Figure 4, it was assumed that subsystems S_1, S_2, and S_3 have slow-response transients, and subsystems S_4 and S_5 have fast-response transients. In order to obtain a reduced-order model, it was necessary to apply a singular perturbation on this decentralized system. The reduced-order model is an approximation of slow-order subsystems S_1, S_2, and S_3. The controller for this reduced-order model was designed using a decentralized control and then implemented to a full-order model. The obtained controller was substituted back into the system equations of an open-loop full-order system in order to obtain the closed-loop system equations. With the closed-loop systems equations, the performance of the system was analyzed. Although this performance was not as good as the one obtained from decentralized control on a full-order system, the computational costs and controller implementation costs were reduced considerably, but with a price of losing some accuracy.

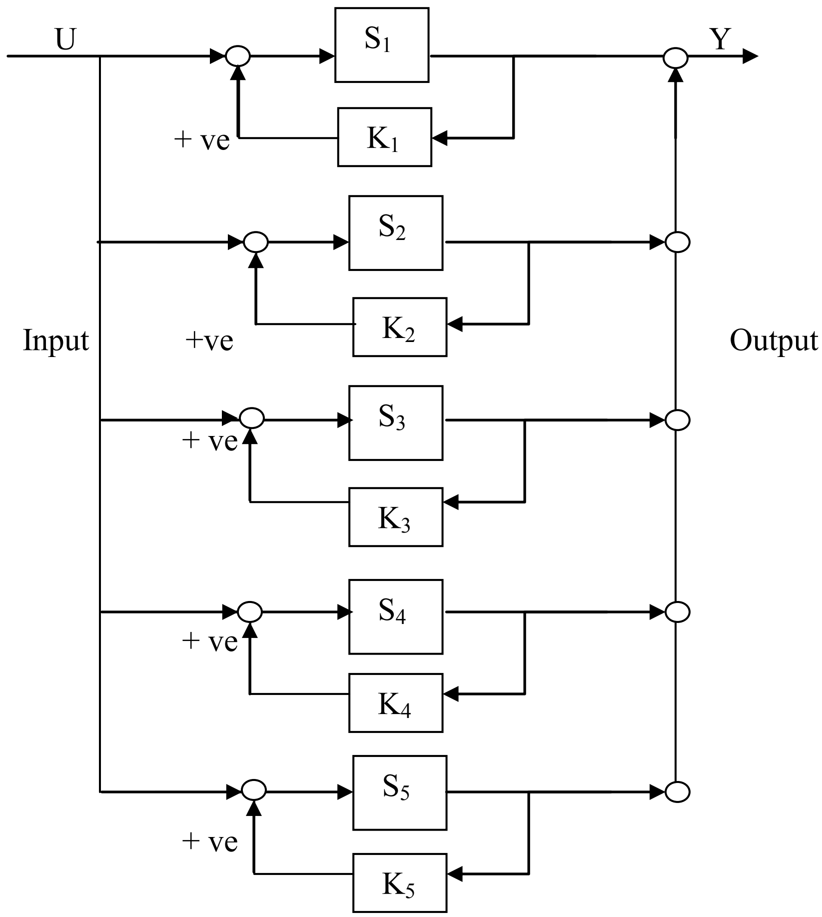

Figure 4: Decentralized singularly perturbed system

CHAPTER 3

DECENTRALIZED CONTROL FOR FULL-ORDER SYSTEM

3.1 Introduction

In the process of designing a controller for a decentralized structure, three problems arose. The first problem was decentralized stabilization. This means that after designing a feedback control, all poles of the closed-loop system must be in the left half plane, i.e., stable. The other problems that arose were decentralized robust control and stochastic decentralized control. A system was considered as having robust control, if the control resulted in asymptotic stability and provided regulation, even under perturbed conditions of the plant. The concept behind stochastic decentralized control is that each sensor connected to each local controller shares some information with other controllers. In this chapter, design of the control that gives decentralized stabilization is considered. Also, the conditions under which the stabilization of the system with the designed decentralized control was possible are presented. The stabilization of the system was achieved via multilevel control in the following sections. It was also possible to achieve stabilization via dynamic compensation, which is an earlier concept.

3.2 Design of Controller

The stabilization of a decentralized control system that uses local and global controllers is explained in this section. An algorithm that explains the stabilization procedure is provided at the end of the section. First, consider a large-scale linear system G(s), which is described by the state equations (3.1) and (3.2).

$$\dot{X}(t) = A\,X(t) + B\,U(t) \tag{3.1}$$

$$Y(t) = C\,X(t) \tag{3.2}$$

This system is decentralized with n linear time invariant subsystems, described by state equations (3.3) and (3.4).

$$\dot{X}_i(t) = A_i\,X_i(t) + B_i\,U_i(t) + \sum_{r=1, r\neq i}^{n} (A_{ir}X_r(t)) \tag{3.3}$$

$$Y_i(t) = C_i\,X_i(t) \tag{3.4}$$

where realization for $G_i(s)$ of each individual subsystem is given as

$$\dot{X}_i(t) = A_i\,X_i(t) + B_i\,U_i(t) \tag{3.5}$$

$$Y_i(t) = C_i\,X_i(t) \tag{3.6}$$

The term $\sum_{r=1, r\neq i}^{n} (A_{ir}X_r(t))$ denotes the interrelationship of the subsystems. The main goal was to design a state feedback controller for the decentralized system. It was assumed that the matrices A_i, B_i, C_i, A_{ir}, and vectors X_i, U_i, and Y_i had appropriate dimensions. The pairs (A_i, B_i) of each subsystem were assumed to be controllable for required pole placement. As required to design a multilevel decentralized control, the control of ith subsystem $U_i(t)$ is

$$U_i(t) = U_{il}(t) + U_{ig}(t) \tag{3.7}$$

where U_{il} is the local control and U_{ig} is the global control as follows:

$$U_{il}(t) = -\sum_{r=1, r\neq i}^{n} (K_{ii}\,X_i(t)) \tag{3.8}$$

$$U_{ig}(t) = -\sum_{r=1, r\neq i}^{n} (K_{ir}\,X_r(t)) \tag{3.9}$$

Substituting the controls given by equations (3.8) and (3.9) into equation (3.3), the closed-loop system for the ith subsystem is obtained with the state equation (3.10):

$$\dot{X}_i(t) = (\,A_i - B_i\,K_i\,)\,X_i(t) + \sum_{r=1, r\neq i}^{n} (\,(\,A_{ir} - B_i\,K_{ir}\,)\,X_r(t)) \tag{3.10}$$

The complete system state vectors are given as $\dot{X}(t) = [\;\dot{X}_1(t)^T\;\dot{X}_2(t)^T \ldots\ldots\ldots \dot{X}_n(t)^T\;]^T$. Since the pair ($A_i$, B_i) is controllable for all subsystems, it is always possible to find feedback gain K_i such hat a set of eigenvalues is assigned to the closed-loop system (A_i-B_i K_i). To stabilize the large-scale system, initially the subsystem needed to be stabilized. The individual stability properties of each subsystem were then aggregated into a single scalar Lyapunov function, and stabilization of large-scale system was attained. Illustrated below is the algorithm that explains the procedure to stabilize the system based on multilevel control [2]:

STEP 1:

The system given in equation (3.1) should be in canonical form, which assures that the pair (A,B) is controllable, which leads to the system given in equation (3.3) being in canonical form, such that pair (A_i, B_i) is controllable for all subsystems. This form is otherwise known as the input decentralized form. If the system is not in required form, a canonical transformation is applied to obtain the system in a canonical representation. In this case, assume that the systems given in both equations (3.1) and (3.3) are in canonical forms.

STEP 2:

In this step, the feedback gains, K_i, for each subsystem must be found. As for the ith subsystem, the pair (A_i, B_i) is controllable, and the resulting closed-loop system equation including controller is

$$\dot{X}_i(t) = (A_i - B_i K_i) X_i(t) \tag{3.11}$$

The characteristic equation of the system obtained is given as det (s I – A_{ci}) = 0, where A_{ci} = ($A_i - B_i K_i$). Comparing this equation with the characteristic equation of the subsystem with desired pole placement, the feedback gain K_i can be solved. Similarly, feedback gains K_i for i = 1, 2…….,n are calculated for all subsystems.

STEP 3:

The aggregation of subsystems to obtain a large-scale interconnected system depends on the aptness of choosing a Lyapunov function, in order for that transformation is to be applied to decoupled closed-loop subsystems. Let M_i be the transformation matrix, and let Λ_i be the block diagonal matrix having poles of the system as block diagonal elements given as

$$\Lambda_i = \text{Block diag} \left\{ \begin{bmatrix} -\lambda_{i1} & w_{i1} \\ -w_{i1} & -\lambda_{i1} \end{bmatrix} \begin{bmatrix} -\lambda_{ij} & w_{ij} \\ -w_{ij} & -\lambda_{ij} \end{bmatrix} \quad -\lambda_{i(j+1)}, \ldots\ldots\ldots -\lambda_{i(ni-j)} \right\}$$

Let the required locations where poles need to be placed for the ith system be $(-\lambda_{i1} \pm w_{i1j}, -\lambda_{i2} \pm w_{i2j}, \ldots\ldots, -\lambda_{ij} \pm w_{ijj}, -\lambda_{i(j+1)}, \ldots\ldots\ldots -\lambda_{i(ni-j)})$, and let the eigenvectors of the closed-loop matrix be A_{ci} be E_{i1}, E_{i2},…..E_{in}. Then the modal matrix M_i will be

$$M_i = [\, \mathrm{Re}(E_{i1}) \mid \mathrm{Im}(E_{i1}) \mid \ldots\ldots \mathrm{Re}(E_{in}) \mid \mathrm{Im}(E_{in})] \tag{3.12}$$

Once the transformation matrix is obtained, the transformed matrices B_i' and A_{ir}' are calculated with equation (3.13):

$$B_i' = M_i^{-1} B_i \text{ and } A_{ir}' = M_i^{-1} A_{ir} M_r \tag{3.13}$$

STEP 4:

In step 3, the local feedback gains, K_i, of each local subsystem have been calculated. Now it is necessary to evaluate the feedback gains of the interconnected systems. Once both the local feedback gains, K_i, and interconnection gains, K_{ir}, are obtained, the system state equations of the overall decentralized system are deduced. The interconnection gains are obtained from equation (3.14), assuming that $(B_i'^T B_i')$ is always invertible.

$$K_{ir}' = [\, (B_i'^T B_i')^{-1} B_i'^T A_{ir}' \,]^T \tag{3.14}$$

Since the values of B_i' and A_{ir}' can be obtained from equation (3.13), substituting those values into equation (3.14) allows the K_{ir}' values to be determined. The decentralized large-scale system is obtained by substituting these feedback gains into the transformed state equation given by

$$\dot{X}_i'(t) = A_i' X_i'(t) + \sum_{r=1, r\neq i}^{n} (A_{ir}' - B_i' K_{ir}') X_r'(t) \tag{3.15}$$

Hence, the overall system ends up with the following state equation for each i_{th} subsystem:

$$\dot{X}_i'(t) = \Lambda_i X_i'(t) + [I_i - B_i'(B_i'^T B_i')^{-1} B_i'^T] \sum_{r=1, r\neq i}^{n} (A_{ir}' X_r(t)) \tag{3.16}$$

STEP 5:

Although a decentralized control was designed, it cannot be assured that the resulting system is stable. Hence, the stability of the whole system must be checked. For the i_{th} subsystem, consider a parameter α_i = min (λ_{ip}) for all possible values of p. The Lyapunov function for the i_{th} subsystem, V_i, is used in such a way that it satisfies the condition

$$V_i (X_i') = (X_i'^T P_i' X_i')^{1/2}, \text{ where } P_i'\Lambda_i + \Lambda_i^T P_i' + H_i' = 0 \tag{3.17}$$

where $P_i' = \beta_i I_i$, $H_i' = 2 \beta_i (\lambda_{i1}, \lambda_{i2}, \ldots\ldots \lambda_{i(j+1)}, \ldots\ldots\ldots \lambda_{i(ni-j)})$, and β is an arbitrary positive constant. Hence, for the entire system, the aggregate Lyapunov function will be $V = (V_1, V_2, V_3 \ldots\ldots\ldots V_n)^T$. The stability condition can be derived by using the vector version of comparison principle, and then the Sevastyanov-Koteliansky condition can be applied to it. If the condition fails to be satisfied, it means that the Lyapunov function is not appropriate.

STEP 6:

If the choice of the Lyapunov function is appropriate, then the overall system turns out to be stable, whereas if the Lyapunov function does not satisfy stability condition, then the value of α_i needs to be reconsidered. The value of α_i, given by the equation provided in step 5, takes the minimum of all real parts of the eigenvalues of the subsystem. Since it did not work out properly, the minimum real value can be excluded, and the next minimum value can be taken as α_i of the real parts of all eigenvalues. Having the new α_i' value, the stability conditions are verified. This process is repeated for larger values of α_i until stabilization is achieved.

3.3 Implementation and Performance Analysis

Once the local feedback gains of each individual subsystem are obtained, it is necessary to implement the local controllers to the full-scale system. For a decentralized system, the controller of the overall system is given by a block diagonal matrix, as discussed in Chapter 2. For the model represented by equations (3.1) and (3.2), let K be the decentralized controller for the overall system. From section 3.2, the local controller gains of each subsystems are obtained as K_1, K_2, ……,K_n. Now the controller K is given as

$$K = \begin{pmatrix} K_1 & \cdots & 0 \\ \vdots & K_2 \ddots & \vdots \\ 0 & \cdots & K_n \end{pmatrix}$$

Once the state feedback controller is obtained, performance analysis can be done on the overall system. The closed-loop subsystem of the large-scale system is given as

$$\dot{X}'(t) = (A - B\,K)\,X'(t) \tag{3.18}$$

To obtain the pole placement of closed-loop system, it is necessary to solve the characteristic equation given as $\det(s\,I - (A\text{-}B\,K)) = 0$. The roots obtained are the resultant pole placements of the system, including the controller. Comparing this pole placement with the desired pole placement, it can be verified whether or not the controller served the purpose of pole placement while stabilizing the entire system at the same time.

3.4 Numerical example

The following example [2] illustrates the scheme of decentralized control, which also stabilizes the entire system. A fifth-order interconnected system has state equation (3.19). Since the control used in the design is the only state feedback control, the matrix C can be anything and does not affect the controller K.

$$\dot{X} = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0.1 & 1 \\ 4 & -1 & 2 & 0 & 0.5 \\ 0.4 & 0.2 & 0 & 0 & 1 \\ 0.5 & 0.2 & 1 & -1 & 2 \end{pmatrix} X + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix} U \tag{3.19}$$

Since the eigenvalues of this system are (3.5, 0.47 $\pm$ j1.56, -0.21 $\pm$ j0.6), the system is not stable. By using the algorithm mentioned in section 3.2, it can be stabilized. The large-scale system given in equation (3.19) is an interconnected system of two subsystems, one of order 3 and another of order 2. System equations of the individual subsystems are as follows:

$$\dot{X}_1 = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 4 & -1 & 2 \end{pmatrix} X_1 + \begin{pmatrix} 1 & 1 \\ 0.1 & 1 \\ 0 & 0.5 \end{pmatrix} X_2 + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{pmatrix} U_1 \tag{3.20}$$

$$\dot{X}_2 = \begin{pmatrix} 0.4 & 0.2 & 0 \\ 0.5 & 0.2 & 1 \end{pmatrix} X_1 + \begin{pmatrix} 0 & 1 \\ -1 & 2 \end{pmatrix} X_2 + \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} U_2 \tag{3.21}$$

where $U_1 + U_2$ are the local controls. The desired pole placement for the subsystem shown by equation (3.20) is (-10, -5 $\pm$ j2), and the desired pole placement for the subsystem shown by equation (3.23) is (-2$\pm$j1). By solving for local feedback gains using state feedback control laws, the obtained feedback gains are K_1 = (294 128 22) and K_2 = (4 6), for the desired pole placements of the two subsystems. The eigenvectors are obtained as E_{v1} and E_{v2}, and the transformation matrices M_1 and M_2 are obtained by using equation (3.12).

$$E_{v1} = \begin{bmatrix} 0.0245+j0.0234 & -0.00990 \\ -0.1694-j0.0678 & 0.0995 \\ -0.9826 & -0.995 \end{bmatrix} \qquad E_{v2} = \begin{bmatrix} -0.4+j0.2 \\ -1 \end{bmatrix}$$

$$M_1 = \begin{bmatrix} 0.0245 & 0.0234 & -0.0099 \\ -0.1694 & -0.0678 & 0.0995 \\ 0.9826 & 0 & -0.9950 \end{bmatrix} \qquad M_2 = \begin{bmatrix} -0.3651 & 0.1826 \\ 0.9129 & 0 \end{bmatrix}$$

Once the transformation matrices of each subsystem are obtained, the transformed matrices B_1', A_{12}', B_2', and A_{21}' need to be found by equations (3.22) and (3.23):

$$B_1' = M_1^{-1} B_1 \text{ and } A_{12}' = M_1^{-1} A_{12} M_2 \tag{3.22}$$

$$B_2' = M_2^{-1} B_2 \text{ and } A_{21}' = M_2^{-1} A_{21} M_1 \tag{3.23}$$

Substituting the appropriate values in equations (3.22) and (3.23), the transformed matrices are obtained, and by putting those matrices in the transformed state equations, the state equations of the interconnected subsystems can be obtained. The system is not yet complete since the global controls U_{ig}' have been found. Using equation (3.14), K_{12} and K_{21} can be determined. Since the transformed matrices are already determined, plugging those values into equation (3.14) gives $K_{12}' = [31.2141 \quad 6.9872]$ and $K_{21}' = [0.9129 \quad -0.0103 \quad -0.9482]$. Using both local and global controls, the overall system is obtained from equation (3.16). Hence the decentralized control system in transformed coordinates will be given as

$$\begin{bmatrix} \dot{X}_1' \\ \dot{X}_2' \end{bmatrix} = \begin{bmatrix} -5.0000 & 2.0000 & 0.0000 & -9.8594 & -1.8118 \\ -2.0000 & -5.0000 & -0.0000 & 42.8077 & 11.9436 \\ -0.0000 & -0.0000 & -10.0000 & 21.1757 & 5.2331 \\ 0.0527 & 0.0092 & -0.0349 & -2.0000 & -1.0000 \\ -0.0264 & -0.0046 & 0.0174 & 1.0000 & -2.0000 \end{bmatrix} \begin{bmatrix} X_1' \\ X_2' \end{bmatrix}$$

To obtain a system with the controller in its original coordinates, the equations needed are as follows:

$$A_{12} = M_1 A_{12}' M_2^{-1}, \qquad A_{21} = M_2 A_{21}' M_1^{-1}, \qquad B_1 = M_1 B_1', \qquad B_2 = M_2 B_2'$$

To do the performance analysis, the poles of the obtained closed-loop system need to be compared to those of the desired pole placement. Poles of the closed-loop system with a controller are ($-5.0614 \pm j2.2034$, $-1.9739 \pm j0.8584$, -9.9296), which are pretty close to the desired pole placement at ($-5 \pm j2$, $-2 \pm j1$, -10).

The example discussed can be solved by hand because of the small order of the interconnected system. If the system is of a much higher order, then the calculations turn out to be difficult. Hence, a MATLAB program can be developed and used to design the decentralized control. The MATLAB code for the example discussed has been provided in Appendix A.

3.5 Summary

A large-scale interconnected system was considered and represented in decentralized form, which is also known as canonical form. For state feedback control, local and global controls were designed and implemented on the full-scale system. The control was designed in such a way that the stabilization was attained even in multilevel control. Once the large-scale decentralized control was designed, the pole placement of a closed-loop system was found and then compared with the desired pole placement. Thus, the performance of the designed decentralized control was analyzed. A MATLAB program was used to design the control as an example. This program helps in solving control-design problems for higher-order systems.

CHAPTER 4

CONTROLLER DESIGN USING SINGULAR PERTURBATION TECHINIQUE

4.1 Introduction

This chapter discusses the design of a controller for a system using the singular perturbation technique. The main purpose of this method was to obtain some approximation at the cost of some accuracy loss. But it is beneficial because of the decrease in the cost of computational procedures and also the reduced cost of controller implementation. The cost of the controller was reduced because in this method, the control was designed for only a few states rather than each and every state. The following sections explain how the time-scale separation among states gives way to the reduced-order model. Also presented are the steps used to obtain a reduced-order model. Once the reduced-order model is in place, two kinds of control can be designed: state feedback and output feedback. The controller was implemented to a full-order model to check for efficiencies of both types of control. The performance of both control systems was analyzed and compared. Based on intuition, it seems that state feedback control should prove to be more accurate since output feedback uses the entire output as feedback and does not have access to individual internal states. In state feedback control, the feedback is taken from required internal states, and therefore, the system will have more accurate pole placement as required. Hence, when the controller design of a numerical example is considered, it can be observed whether or not state feedback gives better performance than output feedback. The performance of a designed controller in this method mainly depends on the value of ε, which is the singular perturbation parameter.

4.2 Slow- and Fast-State Transients

Most full-order systems in practical scenarios have both slow and fast transients. However, the system representation may not be in a standard form that differentiates time scales. If the system is represented in a non-standard form where dividing the slow and fast transients is difficult, it is necessary to perform some existing operations [3] to convert it into standard form. Once the system is in standard form, the eigenvalues of the given system matrix must be determined. Those eigenvalues located nearest the imaginary axis are considered the dominant poles. Transient vectors associated with dominant poles have smaller values because the dominant poles take longer to reach the origin. Those eigenvalues located farthest from the imaginary axis are weak poles, having very high values for their transient vectors. This is because the transient vectors are inversely proportional to the distance of the poles from the imaginary axis. Hence, the fast transient states die sooner than the slow transient states. The state vectors, thus divided into slow- and fast-response state vectors, are shown differently in representation by multiplying the fast-response transients with a very small parameter ε. When the very fast transient state is multiplied by a very small positive scalar ε, the result would be a finite value. Based on these states, a reduced-order model was obtained, as explained in the following sections.

4.3 Reduced-Order Model

A reduced-order model was obtained by approximation of the slow transient state vectors, which are also known as quasi-steady state vectors. Below are the steps to obtain a reduced-order model from a singular perturbation model. A model representation of a system is given as

$$\dot{X} = A\,X + B\,U \tag{4.1}$$

$$Y = C\,X + D\,U \tag{4.2}$$

This model in explicit state variable form is presented as

$$\dot{x} = A_{11}\,x + A_{12}\,z + B_1\,u_1 \tag{4.3}$$

$$\varepsilon\,\dot{z} = A_{21}\,x + A_{22}\,z + B_2\,u_2 \tag{4.4}$$

$$y = C_1\,x + C_2\,z + D\,u \tag{4.5}$$

where x and z are state vectors, u is a control vector, $x \in R^n$, and $z \varepsilon R^m$. The symbol ε denotes a small parameter that can be neglected. In the above model, the state vector $\dot{x}$ has dominant behavior, being a slow-response transient, and $\varepsilon\,\dot{z}$ has a weak effect on the system since it is a fast-response transient. When $\varepsilon = 0$, the order reduction is known as singular perturbation. Thus, differential equation (4.4) is transformed into algebraic equation (4.6) as

$$0 = A_{21}\,x + A_{22}\,z + B_2\,u_2 \tag{4.6}$$

The root of equation (4.6) is $z = -\,A_{22}^{-1}\,(A_{21}\,x + B_2)$ and when substituted back into equations (4.3) and (4.5) yields reduced-order model equations (4.7) and (4.8) as

$$\dot{x}' = (A_{11} - A_{12}\,A_{22}^{-1}\,A_{21})\,x' + (B_1 - A_{12}\,A_{22}^{-1}\,B_2)\,u' \tag{4.7}$$

$$y' = (C_1 - C_2\,A_{22}^{-1}\,A_{21})\,x' + (D - C_2\,A_{22}^{-1}\,B_2)\,u' \tag{4.8}$$

Thus, the order of the reduced-order system will be n, whereas the order of the full-order system will be (n + m).

4.4 Controller Design

4.4.1 State Feedback

Consider the reduced-order model, as shown in Figure 5, obtained from the previous section having system equations (4.7) and (4.8). Let the system matrices be G_a, G_b, G_c, and G_d, given as

$$G_a = (A_{11} - A_{12}\,A_{22}^{-1}\,A_{21})$$

$$G_b = (B_1 - A_{12} A_{22}^{-1} B_2)$$

$$G_c = (C_1 - C_2 A_{22}^{-1} A_{21})$$

$$G_d = (D - C_2 A_{22}^{-1} B_2)$$

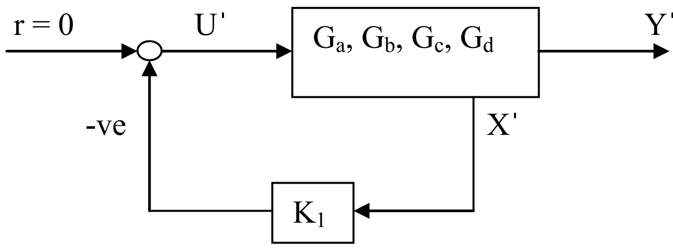

Figure 5: Reduced-order state feedback control

Hence, the reduced-order system is given by

$$\dot{X}' = G_a X' + G_b U' \qquad (4.9)$$

$$Y' = G_c X' + G_d U' \qquad (4.10)$$

where the order of the reduced-order model is n, and the desired pole placement is at locations $(\lambda_1, \lambda_2, \ldots \lambda_n)$. Since this system has state feedback, the feedback control is given by $U' = -K_1 X'$. Substituting this control law in equations (4.9) and (4.10) gives the closed-loop system as

$$\dot{X}' = (G_a - G_b K1) X' \qquad (4.11)$$

Hence, the system matrix is $A_s = (G_a - G_b K_1)$, and the characteristic equation of the system is given by $\det(s I_{nxn} - A_s) = 0$. The eigenvalue matrix λ is given as λ = diagonal matrix $(\lambda_1 \ \lambda_2, \ldots \ \lambda_n)$. The characteristic equation of the system with desired pole placement will be $\det(s I_{nxn} - \lambda) = 0$. By comparing the two characteristic equations in hand, the value of state feedback gain can be calculated. Thus, the state feedback control given by the control law $U' = -K_1 X'$ is achieved.

4.4.2 Output Feedback

Here there will be output feedback given to the system with no external input to the plant. The output feedback law is given by $U' = K_2 Y'$. The system matrices G_a, G_b, G_c, and G_d are those of the reduced-order model, as shown in Figure 6, given by equations (4.7) and (4.8). Substituting the output control law in state and output equations of the reduced-order model, the following equations are obtained:

$$Y' = G_c X' + G_d K_2 Y'$$

$$Y' = (1- G_d K_2)^{-1} G_c X'$$

$$U' = K_2 (1- G_d K_2)^{-1} G_c X'$$

$$\dot{X}' = (G_a + G_b K_2 (1- G_d K_2)^{-1} G_c) X' \tag{4.12}$$

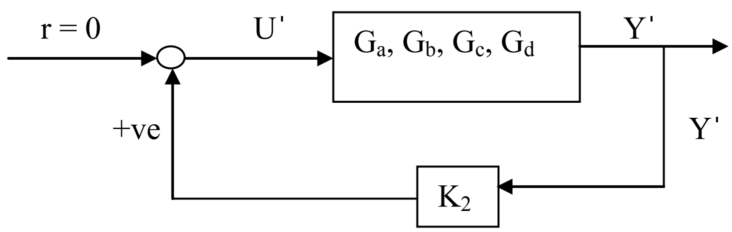

Figure 6: Reduced order output feedback control

The system matrix A_o is given by $A_o = (G_a + G_b K_2 (1- G_d K_2)^{-1} G_c)$. The characteristic equation of the closed-loop system is det(s $I_{nxn} - A_o$)=0. This equation needs to be compared with the characteristic equation of the system with desired pole placement. The characteristic equation of the desired system was presented in the previous section as det (s I - λ) = 0. Comparing these two characteristic equations numerically by hand is not simple. Hence, a MATLAB code was used to solve the feedback gain, K_2. By knowing the feedback gain, K_2, the output feedback control can be provided to the system.

4.5 Implementation to Full-Order Model

The designed control, either state feedback controller or output feedback controller, will be on a smaller order than the controller designed for the full-order system. To obtain the performance of the system with the controller designed using a singular perturbation technique, first it is necessary to obtain the full-order controller from it. From the full-system controller, the closed-loop system of a large-scale system can be represented. If the controller of the full-order system K is of order lx(n + m) and the controller for the reduced-order model (either K_1 or K_2) is only of order pxn, then it is necessary to implement it as a full-order system by padding a sufficient number of zeroes to obtain the K matrix with l row and (n + m) columns. The designed controller (K_1 or K_2) is then placed as the first-block element. Let K_s and K_o be the full-order controllers for the state feedback system and output feedback system, respectively, given as

$$K_s = \begin{bmatrix} K_1 & 0 \\ 0 & 0_{(lx(n+m))} \end{bmatrix} \qquad K_o = \begin{bmatrix} K_2 & 0 \\ 0 & 0_{(lx(n+m))} \end{bmatrix} \tag{4.13}$$

4.6 Comparison between State Feedback Control and Output Feedback Control

Previously in this chapter, two feedback controls for the same desired pole placement were designed. To compare their performances, it is necessary to verify the proximity of the closed-loop system poles to that of desired pole placement. For state feedback control, the closed-loop system was given as $\dot{X} = (A\text{-}B\ K_s)\ X$. Hence, the system matrix is $(A\text{-}B\ K_s)$, and eigenvalues of this matrix provided the poles of the closed-loop system. Similarly, for the output feedback control system, the closed-loop system was $\dot{X} = (A+B\ K_o\ (I\text{-}D\ K_o)^{-1}\ C)\ X$. Since the system matrix was $(A+B\ K_o\ (I\text{-}D\ K_o)^{-1}\ C)$, its eigenvalues provided the poles of the closed-loop system. These two sets of poles for the state and output feedback control systems were evaluated with the poles desired to be placed. The error in two cases was calculated and analyzed. Of these two controls, it is obvious that state feedback control provided better performance, since the individual states can be accessed and given as feedback according to requirements. Output control provided less accuracy in performance, in comparison, and this is illustrated with an example in the next section.

4.7 Numerical Example

In order to explain the process of designing a controller using the singular perturbation technique, the following example [3] is used. A system having both slow and fast state transients is represented by state equations (4.14) and (4.15):

$$\begin{bmatrix} \dot{X} \\ \varepsilon \dot{Z} \end{bmatrix} = \begin{bmatrix} -0.2000 & 0.2000 & 0 & 0 \\ 0 & -0.5000 & 0.5000 & 0 \\ 0 & 0 & 0 & 1.0000 \\ 0 & 0 & -1.0000 & -2.0000 \end{bmatrix} \begin{bmatrix} X \\ Z \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} U \tag{4.14}$$

$$Y = [\,1 \quad 0 \quad 0 \quad 0\,] \begin{bmatrix} X \\ Z \end{bmatrix} \tag{4.15}$$

where $X \in R^2$ and $Z \varepsilon R^2$ are the slow- and fast-response states, respectively, thus dividing the full-order system into two subsystems. The individual subsystems are given by equations (4.16), (4.17) and (4.18):

$$\dot{X} = \begin{bmatrix} -0.2000 & 0.2000 \\ 0 & -0.5000 \end{bmatrix} X + \begin{bmatrix} 0 & 0 \\ 0.5 & 0 \end{bmatrix} Z + \begin{bmatrix} 0 \\ 0 \end{bmatrix} U_1 \tag{4.16}$$

$$\varepsilon \dot{Z} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} X + \begin{bmatrix} 0 & 1.0000 \\ -1.0 & -2.0000 \end{bmatrix} Z + \begin{bmatrix} 0 \\ 0 \end{bmatrix} U_2 \tag{4.17}$$

$$Y = [\,1 \quad 0\,]\, X + [\,0 \quad 0\,]\, Z \tag{4.18}$$

Setting $\varepsilon = 0$ in equation (4.17) and then substituting the root for Z from equation (4.17) back into equations (4.16) and (4.18), the reduced-order model was obtained. Thus, the obtained system matrices of the reduced-order model are given as

$$G_a = \begin{bmatrix} -0.2 & 0.2 \\ 0 & -0.5 \end{bmatrix} \qquad G_b = \begin{bmatrix} 0 \\ 0.5 \end{bmatrix} \qquad G_c = [\,1 \quad 0\,] \qquad G_d = 0$$

Let the desired pole placement be at the locations (-0.3, -0.4). For state feedback control, let K_1 be the feedback gain of the control system. Using the MATLAB command ―place," the gain K_1 is found using G_a and G_b. The obtained feedback gain is K_1 = [0.2 0]. This is implemented in the full-order model control K_{p1} by padding zeroes. Hence, K_{p1} is given as K_{p1} = [0.2 0 0 0]. The same system has been used to design a controller with two different ε values to see how the accuracy of the reduced-order model design depends on the parameter of singular perturbation. The poles are chosen arbitrarily for the closed-loop system. When this controller is used to obtain a closed-loop system in two cases—one having ε = 0.1 and the other having ε = 0.01—the following are the pole placements of closed-loop system:

For ε = 0.1:

Pole placement from full-order controller design is [-10.1 -10 -0.3 -0.4]

Pole placement from reduced-order model design is [-10.0022 ± 0.1465i, -0.3157, -0.38]

For ε = 0.01:

Pole placement from full-order controller design is [-101 -100 -0.3 -0.4]

Pole placement from reduced-order model design is [-100.0±0.1400i, -0.30, -0.40]

For output feedback control, let K_2 be the gain of reduced-order output feedback control. Using the MATLAB program provided in Appendix 3, the gain K_2 is found using G_a, G_b, G_c, and G_d as K = [0 -0.88]. This can be implemented in the full-order system by padding zeroes. The controller K_{p2} of the full-order system for output feedback is obtained as K_{p2} = [0 -0.88 0 0]. This system has been checked for controller design using two different values of ε: 0.1 and 0.01.

The pole placements obtained for the closed-loop system, when considering different values for ε, are as follows:

For ε = 0.1:

Pole placement from full-order controller design is [-10.1 -10 -0.3 -0.4]

Pole placement from reduced-order model design is [$-10.2224 \pm 2.0916i$, -0.2, -0.05]

For ε = 0.01:

Pole placement from full-order controller design is [-101 -100 -0.3 -0.4]

Pole placement from reduced-order model design is [$-100.22 \pm 6.63i$, -0.2, -0.06]

For state feedback control, from the obtained results, it is observed that the design when ε = 0.01 gives better performance than the design when ε = 0.1. For output feedback control, from the resulting pole placements presented, it is once again clear that the design when ε = 0.01 gives performance than when ε = 0.1. Hence, from these two cases, it is evident that reduction of the ε value in the system gives better performances while designing the controller using a singular perturbation technique. On the other hand, when pole placement of the state feedback control is compared with that of output feedback control, the comparison showed that pole placement error for output feedback is large when compared to that of state feedback pole placement. Hence, it is also proven that state feedback is more useful that output feedback. The MATLAB program codes to solve the controller design for a singularly perturbed system as explained in the example are given in Appendices B and C. Appendix B shows the program used for controller design using state feedback control, and Appendix C shows the program used to design a controller for the output feedback control system.

4.8 Summary

In this chapter, the complete designing process of a controller for full-order system was presented. This design mainly depends on the singular perturbation technique. A controller was designed for both state feedback control and output feedback control systems. The design was done for different values of ε, a parameter of singular perturbation, and the dependency of accuracy on the value of ε was verified. A comparison was also done on performances of the state feedback control and the output feedback control to determine which gave the desired performance for a system. A numerical example was given to explain the procedure in detail. A MATLAB program code was written for both state and output feedback controls to facilitate the design process. In this way, the controller of a singularly perturbed system was designed.

CHAPTER 5

DECENTRALIZED SINGULARLY PERTURBED SYSTEM

5.1 Introduction

Chapters 3 and 4 presented the decentralized control of a system and controller design using a singular perturbation technique. In both procedures, calculations were simplified and cost reduction was attained. To simplify further, both techniques can be combined and implemented on a given system. If there is a large-scale system that is decentralized and also singularly perturbed, both procedures can be applied. First, a reduced-order model of a large-scale system was obtained using singular perturbation. For the reduced-order model, a decentralized controller was designed. The obtained controller was implemented to a full-order model by padding zeroes. Since the approximation was done at two levels, this is a simpler procedure than the other two procedures discussed in earlier chapters. Because the approximation was performed twice, the system lost some information and therefore some accuracy as well. However, a lower price for controller installation and lower computational and storage needs were attained at the cost of this lost accuracy.

5.2 Decentralized Control Design for Full-Order System

In this section, a decentralized singularly perturbed system is considered. A decentralized controller for a full-order system was designed and compared to the controller designed based on the reduced-order model in the later sections. Consider a decentralized singularly perturbed system with the following output and state equations:

$$Y(t) = C\,X(t) \tag{5.1}$$

$$\begin{bmatrix} \dot{X}(t) \\ \varepsilon\,\dot{Z}(t) \end{bmatrix} = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix} \begin{bmatrix} X(t) \\ Z(t) \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} U(t) \tag{5.2}$$

This system is decentralized with (n + m) linear time invariant subsystems where the slow-response part of the system has n subsystems, and the fast-response part of the system has m subsystems. Let $A = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix}$ and $B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}$. To design a controller, a state feedback control was given to the system as $U(t) = K_f \; [\; X^T(t) \quad Z^T(t) \;]^T$, where K_f is the state feedback gain. Substituting this control law into system equations, the closed-loop systems equations are given as

$$\begin{bmatrix} \dot{X}(t) \\ \varepsilon \; \dot{Z}(t) \end{bmatrix} = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix} \begin{bmatrix} X(t) \\ Z(t) \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} K_f \begin{bmatrix} X(t) \\ Z(t) \end{bmatrix}$$

$$\begin{bmatrix} \dot{X}(t) \\ \varepsilon \; \dot{Z}(t) \end{bmatrix} = \left[\begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} K_f \right] \begin{bmatrix} X(t) \\ Z(t) \end{bmatrix}$$

Let ($\lambda_1, \lambda_2, \ldots.. \lambda_n \ldots\ldots \lambda_{(n+m)}$) be the poles to be placed. Using the algorithm discussed in section 3.2 of Chapter 3, the controller was designed. Again, this controller was a block diagonal matrix with a controller for each subsystem as diagonal elements. Let $K_1, K_2, \ldots.. K_n, \ldots\ldots K_{(n+m)}$ be the individual controllers of all the subsystems. Then K_f is given as

$$K = \begin{bmatrix} K_1 & \cdots\cdots & K_{ij} \\ \vdots & K_2 \quad \ddots & \vdots \\ K_{(n+m)1} & \cdots\cdots & K_{(n+m)} \end{bmatrix}$$

where K_{ij} is the interconnected gain for the i_{th} and j_{th} system. K_{ij} can also be evaluated using the algorithm given in section 3.2. Using step 4 of the algorithm, the system matrix of the closed-loop system after including the designed controller was obtained. In this way, both the decentralized controller of the full-order system and the system matrix A_f of the closed-loop system were calculated. System matrix A_f is given by $A_f = [A - B\,K_f]$.

5.3 Decentralized Controller Design Based on Reduced-Order Controller

In this section, a reduced-order model is deduced from a full-order model and a decentralized controller is designed for it. For the system in equations (5.1) and (5.2), $\dot{X}(t)$ represents slow-state vectors, and $\dot{Z}(t)$ represents fast-state vectors. As the system is already in a perturbed model, the singular perturbed technique is applied by making the parameter $\varepsilon = 0$, then the state equations are as follows, assuming that A_4 is always invertible:

$$\dot{X}(t) = A_1\, X(t) + A_2\, Z(t) + B_1\, U_1(t) \tag{5.3}$$

$$0 = \varepsilon\, \dot{Z}(t) = A_3\, X(t) + A_4\, Z(t) + B_2\, U_2(t)$$

Solving for Z(t) and substituting this back into equation (5.3) gives the following reduced-order model:

$$Z(t) = -A_4^{-1}\, A_3\, X(t) - A_4^{-1}\, B_2\, U_2(t)$$

$$\dot{X}(t) = A_1\, X(t) - A_2\, A_4^{-1}\, A_3\, X(t) + B_1\, U_1(t) - A_2\, A_4^{-1}\, B_2\, U_2(t)$$

$$\dot{X}'(t) = (\, A_1 - A_2\, A_4^{-1}\, A_3)\, X'(t) + (\, B_1 - A_2\, A_4^{-1}\, B_2\,)\, U'(t)$$

The above system is the state equation of reduced-order model having n individual states. Once the model is in place and since it is a decentralized system, the decentralized controller can be designed using the algorithm shown in section (3.2). Let $K_{r1}, K_{r2}, K_{r3} \ldots K_{rn}$ be the local state feedback gains and K_{rij} be the interconnected feedback gain for i_{th} and j_{th} local subsystems. All the gains are found using the algorithm, and then the final controller K_{rp} is obtained as

$$K_{rp} = \begin{pmatrix} K_{r1} & \cdots & K_{ij} \\ \vdots & K_{r2} & \vdots \\ K_{ij} & \cdots & K_{rn} \end{pmatrix}$$

This is the decentralized controller for the reduced-order system.

5.4 Implementation and Performance Analysis

The decentralized controller obtained in the previous section is the controller for only the N_{th}-ordered reduced model**.** To obtain the full-order decentralized controller, it is necessary to implement the controller K_{rp} on the full-order model. This is achieved by padding zeroes. Let the decentralized controller for the full-order system be K_r. Based on the reduced-order model,

$$K_r = \begin{bmatrix} K_{rp} & \cdots\cdots & 0 \\ \vdots & & \vdots \\ 0 & \cdots\cdots & 0 \end{bmatrix}$$

This K_r is an appropriate order related to A and B. The closed-loop system is obtained from the controller by substituting the control law in equation (5.1) as

$$\dot{X}(t) = (A\text{-}B\ K_r)\ X(t).$$

Hence, the system matrix is $A_r = (A\text{-}B\ K_r)$. From section 5.2, the system matrix A_f of the closed-loop system with a full-order decentralized controller was used to verify the sensitivity of the design using the singular perturbation technique and compare the poles of the closed-loop system from the reduced-order model with the poles of the closed-loop system from the full-order model. Hence, the eigenvalues of matrix A_r were compared with the eigenvalues of matrix A_f. In that way the performance of both the systems was analyzed.

5.5 Application to Large-scale Decentralized Power System

The procedure presented in the earlier sections of this chapter was needed to be applied to a real-time system to check the robustness and reliability of the system. Consider a power plant having five substations where two subsystems have slow-response state transients and three subsystems have fast-response state transients. The state equations of the system are given below [5]. In this system, vector X represents slow-response states, and vector Z represents fast-response states.

$$\begin{bmatrix} \dot{X} \\ \varepsilon\dot{Z} \end{bmatrix} = \left[\begin{array}{cc|ccc} -0.5 & 0 & 0.1 & -0.2 & 0.1 \\ 0 & -1 & 0.1 & 0.3 & 0.2 \\ \hline 0.4 & 0.3 & -0.5 & 0 & 0 \\ -0.4 & 0.4 & 0 & -0.45 & 0 \\ 0.35 & 0.3 & 0 & 0 & -0.4 \end{array}\right] \begin{bmatrix} X \\ Z \end{bmatrix} + \left[\begin{array}{c|c} 1 & 0 \\ 0 & 1 \\ \hline 0.4 & 0 \\ 0 & -0.5 \\ 0 & 0.6 \end{array}\right] \begin{bmatrix} U_1 \\ U_2 \end{bmatrix}$$

Again, the slow-order system is a decentralized system with states of two subsystems, and the fast-order system is a decentralized system with states of three subsystems. Now the decentralized controller for a full-order system is designed in following way, using the method discussed in section 5.2. The required pole placement for the system is (-1, -0.5, -4, -4.5, -5). To design a controller for the entire system, let it be divided into two main subsystems and the value of ε be 0.1. The first subsystem has the following matrices for system equations:

$$A_{11} = \begin{bmatrix} -0.5 & 0 \\ 0 & -1 \end{bmatrix} \qquad B_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

The controller for this system is designed using the MATLAB command ―place," and the obtained controller is $\begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}$

Similarly, the other subsystem has the following system matrices, and the given controller is obtained using MATLAB:

$$A_{22} = \begin{bmatrix} -5 & 0 & 0 \\ 0 & -4.5 & 0 \\ 0 & 0 & -4.0 \end{bmatrix} \quad B_2 = \begin{bmatrix} 4 & 0 \\ 0 & -5 \\ 0 & 6 \end{bmatrix} \quad \text{Controller} = \begin{bmatrix} -1.137 & 0.182 & 0.133 \\ 0 & 6.56 & 4.2 \end{bmatrix}$$

Using the eigenvectors, transformation matrices M_1 and M_2 are found and thus the transformed matrices are found. Then the system matrix is obtained as

$$A = \begin{pmatrix} -1 & 0 & 0 & 0 & 0 \\ 0 & -0.5 & 0 & 0 & 0 \\ -0.148 & 0.886 & -5 & 0 & 0 \\ 0.647 & -3.881 & 0 & -4.5 & 0 \\ -0.583 & 3.497 & 0 & 0 & -4.5 \end{pmatrix}$$

The eigenvalues of this matrix are the obtained pole placements given by (-1, -0.5, -0.4, -0.45,-5).

Now it is necessary to obtain a reduced-order model and design a decentralized controller for it to achieve a simplified procedure using the method required as discussed in section 5.3. A MATLAB program code was used to do the calculations, which are provided in Appendix E. The reduced-order model thus obtained is of order 2 x 2 having G_a and G_b:

$$G_a = \begin{bmatrix} -0.1547 & -0.0428 \\ -0.3617 & -0.8233 \end{bmatrix} \qquad G_b = \begin{bmatrix} 1.08 & 0.3722 \\ 0.08 & 0.3667 \end{bmatrix}$$

This reduced-order system is again a decentralized system with two subsystems, each of order 1 x 1. Let the pole placements for the subsystems be (-1) and (-0.5). The individual controllers were obtained, and from these, the decentralized controller K_p of the reduced-order model was obtained as follows:

$$K_1 = \begin{bmatrix} 0.6996 \\ 0.2411 \end{bmatrix} \quad K_2 = \begin{bmatrix} -0.1837 \\ -0.8417 \end{bmatrix} \quad K_p = \begin{bmatrix} 0.6996 & -0.1837 \\ 0.2411 & -0.8417 \end{bmatrix}$$

The required controller for the full-order system should be of order 2 x 5, according to the dimensions of A and B. Hence, a 2 x 3 ordered zero matrix is padded to obtain K as

$$K = \begin{bmatrix} 0.6996 & -0.1837 & 0 & 0 & 0 \\ 0.2411 & -0.8417 & 0 & 0 & 0 \end{bmatrix}$$

Using the control law equation, Ac = A - B K, the system matrix was obtained, and the eigenvalues of the system matrix, which are the pole placements of the system, were found as (-3.7464, -4.6524, -5.0678, -0.6956$\pm$0.5170i). It can be seen from this result that the error in pole placement is 6.2% on average for $\varepsilon = 0.1$, the reduced-order model decentralized controller design is used. The robustness of the designed controller was verified by considering different values of “ε.” Tables 1 and 2 show the desired pole placement, obtained pole placement, and error comparison for various “ε” values considered. Pole placement was chosen arbitrarily.

TABLE 1

POLE PLACEMENTS OBTAINED FOR VARIOUS ε VALUES

Value of ε	**Desired Pole Placement of System**	**Obtained Pole Placement with Controller**	**Error/Deviation of Obtained Poles from Desired Poles**
0.5	-0.8, -0.9, -1, -1, -0.5	-0.825, -1.057, -1.485, -0.346 $\pm$ 0.539i	0.0311, 0.1741, 0.485, 0.36
0.1	-4, -4.5, -5, -1, -0.5	-3.74, -4.65, -5.068, -0.6956 $\pm$ 0.517i	0.065, 0.033, 0.014, 0.1378
0.05	-8, -9, -10, -1, -0.5	-7.69, -9.125, -10.08, -0.729 $\pm$ 0.4674i	0.038, 0.014, 0.008, 0.134
0.01	-40, -45, -50, -1, -0.5	-39.66, -45.1, -50.095, -0.7466 $\pm$ 0.4272i	0.0086, 0.0025, 0.0019, 0.14
0.001	-400, -450, -500, -1, -0.5	-399.65, -450.11, -500.10, -0.75 $\pm$ 0.42i	0.00087, 0.00024, 0.0002, 0.15

TABLE 2

COMPARISON OF ERROR IN POLE PLACEMENT FOR VARIOUS ε VALUES

Value of ε	**Desired Pole Placement**	**Percentage Error In Pole Placement**
0.5	-0.8, -0.9, -1, -1, -0.5	21.1
0.1	-4, -4.5, -5, -1, -0.5	6.2
0.05	-8, -9, -10, -1, -0.5	4.85
0.01	-40, -45, -50, -1, -0.5	3.8
0.001	-400, -450, -500, -1, -0.5	3.6

Thus, it is clear from Table 2 that to obtain an error in pole placement of less than 5%, the ε value should be less than or near equal to 0.05. In this way, the robustness of system is assured. Once the robustness is verified, it is needed to verify the reliability of the system. Reliability in a practical sense means that even if one of the controllers of an individual subsystem or a part of any of the controller fails, the system should not deviate much from the desired performance. To verify reliability, the controller designed is used to find the error in pole placement by making one or more elements of matrix K intentionally zero. Tables 3 and 4 show the tabulated information of pole placements obtained and the deviation of the obtained poles from that of the desired poles for various cases of K, where, in each case, some components of the controller are considered to be failed. The desired pole placement, which acts as reference in calculating percentage error of pole placement, is (-4, -4.5, -5, -1, -0.5).

Table 3 provides the error for each pole placement, depending on the deviation of the obtained poles from the desired ones. Table 4 provides the summary of errors obtained on average in each failure case of controller K.

From the tabular values in Figure 4, it can be clearly seen that when the entire controller of a subsystem is considered, the error percentage is more than that of the controller included. It is also observed that the error percentage of the pole placement when only part of the controller of subsystem is neglected is smaller than that of the pole placement with the total controller excluded from the system. Therefore, it can be concluded that the percentage error in pole placement is more if the entire controller is excluded rather than neglecting part of it. It is also clear that even if a controller or part of it is fails, the percentage error is not more than 13%. Thus, the entire system is reliable, even if some components fail to work.

TABLE 3

POLE PLACEMENTS OBTAINED FOR VARIOUS FAILURE CASES OF K

Failure Cases of Controller K	Pole Placements Obtained with Controller K	Error/Deviation of Obtained poles from Desired Poles
$K = \begin{bmatrix} 0.699 & -0.184 & 0 & 0 & 0 \\ 0.241 & 0.842 & 0 & 0 & 0 \end{bmatrix}$	-3.746, -4.652, -5.0678, -0.6956 $\pm$ 0.517i	0.065, 0.033, 0.014, 0.138
$K = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0.241 & 0.842 & 0 & 0 & 0 \end{bmatrix}$	-3.6739, -4.5925, -5.1455, -0.3732 $\pm$ 0.3426i	0.0815, 0.0205, 0.0291, 0.2467
$K = \begin{bmatrix} 0.699 & -0.184 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$	-3.9685, -5.0639$\pm$0.08i, -0.8016 $\pm$ 0.2562i	0.0078, 0.124, 0.0125, 0.1585
$K = \begin{bmatrix} 0 & -0.184 & 0 & 0 & 0 \\ 0.241 & 0.842 & 0 & 0 & 0 \end{bmatrix}$	-3.6923, -4.6024, -5.1506, -0.3565 $\pm$ 0.4643i	0.0769, 0.0227, 0.03012, 0.17
$K = \begin{bmatrix} 0.699 & 0 & 0 & 0 & 0 \\ 0.241 & 0.842 & 0 & 0 & 0 \end{bmatrix}$	-3.726, -4.6379, -5.0636, -0.7153 $\pm$ 0.3983i	0.0685, 0.0306, 0.01272, 0.181
$K = \begin{bmatrix} 0.699 & -0.184 & 0 & 0 & 0 \\ 0 & 0.842 & 0 & 0 & 0 \end{bmatrix}$	-3.8, -4.7462, -5.0348, -0.6384 $\pm$ 0.4875i	0.05, 0.0547, 0.00696, 0.197
$K = \begin{bmatrix} 0.699 & -0.184 & 0 & 0 & 0 \\ 0.241 & 0 & 0 & 0 & 0 \end{bmatrix}$	-3.9451, -4.9239, -5.1163, 0.8572 $\pm$ 0.2623i	0.0137, 0.094, 0.0233, 0.197

TABLE 4

COMPARISON OF ERROR IN POLE PLACEMENT: DIFFERENT FAILURE CASES OF K

Failure Cases of Controller K	Percentage Error in Pole Placement
$K = \begin{bmatrix} 0.699 & -0.184 & 0 & 0 & 0 \\ 0.241 & 0.842 & 0 & 0 & 0 \end{bmatrix}$	6.24
$K = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0.241 & 0.842 & 0 & 0 & 0 \end{bmatrix}$	12.6
$K = \begin{bmatrix} 0.699 & -0.184 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$	10.61
$K = \begin{bmatrix} 0 & -0.184 & 0 & 0 & 0 \\ 0.241 & 0.842 & 0 & 0 & 0 \end{bmatrix}$	7.5
$K = \begin{bmatrix} 0.699 & 0 & 0 & 0 & 0 \\ 0.241 & 0.842 & 0 & 0 & 0 \end{bmatrix}$	7.32
$K = \begin{bmatrix} 0.699 & -0.184 & 0 & 0 & 0 \\ 0 & 0.842 & 0 & 0 & 0 \end{bmatrix}$	7.71
$K = \begin{bmatrix} 0.699 & -0.184 & 0 & 0 & 0 \\ 0.241 & 0 & 0 & 0 & 0 \end{bmatrix}$	8.2

5.6 Summary

In this chapter, the techniques discussed in earlier chapters were used and implemented to design a controller for a decentralized singularly perturbed system. To have a reference design for comparison purposes, a decentralized control was designed for a full-order system. Later, using singular perturbation, a reduced-order model was found, and a decentralized control was designed for it. Then the reduced model controller was implemented on full-order system and the resulting closed-loop system was found. The performance of this closed-loop system was compared to that of the closed-loop system from a full-order design. This developed procedure was applied on a real-time practical system of a decentralized power system, which was both decentralized and singularly perturbed. Upon application on a power system, it was observed that the system had a controller that was reliable and robust in nature. It was also observed that the computational needs were lessened and the cost of implementing the controller was reduced because of zero elements in the controller matrix, K_r. This system lost some accuracy because of the approximation of the slow-response system.

CHAPTER 6

CONCLUSIONS

6.1 Conclusions

The two methods of obtaining a system controller were discussed. Combining both methods and applying them to a decentralized singularly perturbed power system demonstrated this to be a simplified procedure in many aspects. The designed controller assured a stable pole placement. Robustness of the system was verified by providing a fixed set of values for ε—the parameter of a singular perturbation. The controller was also demonstrated to be reliable by observing the results that even failure of one or more components of a controller resulted in the being less than a prestated percentage. Therefore stability, robustness, and reliability of the system were achieved, although the procedural methods were simplified to a great extent. One of the main advantages of this method is the reduced computational costs since the controller needs to be designed only for the reduced-order model. Another advantage is the cost of controller implementation, since the designed controller has many zero elements in it. For this example, reliability was guaranteed by showing that if the controller from one subsystem fails, the controller from another system acts as back up and serves the purpose. The disadvantage of this method is that because of the aggregation of the slow-response system in the reduced-order model, some accuracy was lost. This in turn resulted in an increase of error in pole placement. But this can be compensated with other benefits. Finally, it can be concluded that this method of controller design for decentralized singularly perturbed systems is efficient enough to serve the desired purpose of reliability and robustness.

6.2 Scope of Future Work

This thesis work presented the controller design for a decentralized singularly perturbed system having state feedback. This work could be extended to output feedback systems. To obtain the optimal controller, the optimal linear quadratic (LQ) design could be used to obtain the Q and R of the equations. Another approach that could be followed would be to use the H_∞ controller method to design the controller for a large-scale system.

REFERENCES

REFERENCES

[1] Petar V. Kokotovic, ―Application of Singular Perturbation Techniques to Control problems," *SIAM Review*, Vol. 26, No. 4, pp. 501-550, October, 1984.

[2] Mohammad Jamshidi, *Large-Scale Systems: Modeling, Control and Fuzzy Logic,* Prentice-Hall, New Jersey, 1997.

[3] Petar V. Kokotovic, Hassan K. Khali, and John O'Reilly, *Singular Perturbation Methods in Control: Analysis and Design,* Academic Press Inc., Florida, 1986.

[4] Bahram Shahian and Michael Hassul, *Control System Design Using Matlab,* Prentice-Hall, New Jersey, 1993.

[5] Kai-Chao Yao, ―Robust Sub-Optimal Pole Assignment of Decentralized Actuator Type Large-scale Systems by the Aggregation Matrix with Robustness Test," *Proceedings of the American Control Conference*, pp. 4506-4511, Denver, Colorado, June 4-6 2003.

APPENDICES

APPENDIX A

MATLAB M-File for Decentralized Controller of Full-Order System

```
% Full-order system

af=[0 1 0 1 1; 0 0 1 0.1 1; 4 -1 2 0 0.5; 0.4 0.2 0 0 1; 0.5 0.2 1 -1 2]
bf=[0 0; 0 0; 1 0; 0 0; 0 1]

a11=af(1:3,1:3)
a12=af(1:3,4:5)
a21=af(4:5,1:3)
a22=af(4:5,4:5)
b1=bf(1:3,1)
b2=bf(4:5,2)

eig(a11)
eig(a22)

p1=[-5+2i -5-2i -10]
p2=[-2+1i -2-1i]

k1=place(a11,b1,p1)
k2=place(a22,b2,p2)

a1=a11-b1*k1
a2=a22-b2*k2
eig(a1)
[V1, D1] = eig(a1)
eig(a2)
[V2, D2] = eig(a2)

M1=[real(V1(1,1)) imag(V1(1,1)) real(V1(1,3));real(V1(2,1)) imag(V1(2,1))
real(V1(2,3)); real(V1(3,1)) imag(V1(3,1)) real(V1(3,3))]
M2=[real(V2(1,2)) imag(V2(1,2)); real(V2(2,2)) imag(V2(2,2))]

af11=(inv(M1))*a1*M1
af22=(inv(M2))*a2*M2
af12=(inv(M1))*a12*M2
af21=(inv(M2))*a21*M1
bf1=(inv(M1))*b1
bf2=(inv(M2))*b2

AF11=af11
AF22=af22
AF12=(eye(3)-bf1*(inv(bf1'*bf1))*bf1')*af12
AF21=(eye(2)-bf2*(inv(bf2'*bf2))*bf2')*af21

A=[AF11 AF12; AF21 AF22]


eig(A)
```

APPENDIX B

MATLAB M-File for Reduced-Order Controller Design for State Feedback

```
% Full-order system
af=[-1/5 1/5 0 0; 0 -1/2 1/2 0; 0 0 0 1; 0 0 -1 -2]
bf=[0; 0; 0; 1]
cf=[1 0 0 0]
df=0
% seperating slow and fast transient states
a11=af(1:2,1:2)
a12=af(1:2,3:4)
a21=af(3:4,1:2)
a22=af(3:4,3:4)
b1=bf(1:2,1)
b2=bf(3:4,1)
c1=cf(1,1:2)
c2=cf(1,3:4)
% making e=0, obtaining reduced-order model / singular perturbation technique
ga=a11-a12*(inv(a22))*a21
gb=b1-a12*(inv(a22))*b2
gc=c1-c2*(inv(a22))*a21
gd=-c2*(inv(a22))*b2
% designing controller for reduced-order model with state feedback
s=[-0.3 -0.4]
k=place(ga,gb,s)
ac=ga-gb*k
%Comparision of pole placement when e=0.1
af1=[-1/5 1/5 0 0; 0 -1/2 1/2 0; 0 0 0 10; 0 0 -10 -20]
bf1=[0; 0; 0; 10]
cf1=[1 0 0 0]
df1=0
eig(af1)
% Implementing controller of reduced-order model to the full-order system
% with e=0.1
Kp=[k 0 0]
ap1=af1-bf1*Kp
% Verifying the pole placement after design using reduced-order model with
% e=0.1
eig(ap1)
% Design of controller for full-order system with e=0.1
o1=[-0.3 -0.4 -10 -10.1]
Ko1=place(af1,bf1,o1)
ao1=af1-bf1*Ko1
eig(ao1)
%Comparision of pole placement when e=0.01
af2=[-1/5 1/5 0 0; 0 -1/2 1/2 0; 0 0 0 100; 0 0 -100 -200]
bf2=[0; 0; 0; 100]
cf2=[1 0 0 0]
df2=0
eig(af2)
% Implementing controller of reduced-order model to the full-order system
% with e=0.1
Kp=[k 0 0]
ap2=af2-bf2*Kp
```

APPENDIX B (continued)

```
% Verifying the pole placement after design using reduced-order model with
% e=0.1
eig(ap2)
% Design of controller for full-order system with e=0.01
o2=[-0.3 -0.4 -100 -101]
Ko2=place(af2,bf2,o2)
ao2=af2-bf2*Ko2
eig(ao2)
% Verifying the pole placement after design using full-order model with
% e=0.1
eig(ao1)
eig(ap1)
% Verifying the pole placement after design using full-order model with
% e=0.01
eig(ao2)
eig(ap2
```

APPENDIX C

MATLAB M-File for Reduced-Order Controller Design for Output Feedback

```
% Full-order system
af=[-1/5 1/5 0 0; 0 -1/2 1/2 0; 0 0 0 1; 0 0 -1 -2]
bf=[0; 0; 0; 1]
cf=[1 0 0 0]
df=0
% seperating slow and fast transient states
a11=af(1:2,1:2)
a12=af(1:2,3:4)
a21=af(3:4,1:2)
a22=af(3:4,3:4)
b1=bf(1:2,1)
b2=bf(3:4,1)
c1=cf(1,1:2)
c2=cf(1,3:4)
% making e=0, obtaining reduced-order model / singular perturbation technique
ga=a11-a12*(inv(a22))*a21
gb=b1-a12*(inv(a22))*b2
gc=c1-c2*(inv(a22))*a21
gd=-c2*(inv(a22))*b2
% designing controller for reduced-order model
s=[-0.3, 0; 0, -0.4]
[ng,dg]=ss2tf(ga,gb,gc,gd)
Dd=[0 1 0.7 0.012]
f=Dd;
m=length(ng)
nn=length(dg)
n=nn-1
n2=2*n
padzero=zeros(1,nn-m)
ngpad=[padzero ng]
s_coll2=flipdim([dg' ngpad'],1)
sylv=zeros(2*n)
for i=1:2:2*n-1,
    bgnrow=(i+1)/2
    sylv(bgnrow:bgnrow+n,i:i+1)=s_coll2
end;
delta=flipdim(Dd',1)
x=sylv\delta;
for i=1:2:2*n-1,
    r=(i-1)/2;
    a1(n-r)=x(i);
    a1=real(a1);
    b1(n-r)=x(i+1);
    b1=real(b1);
end;
a1=real(a1)
b1=real(b1)
[k1a,k1b,k1c,k1d]=tf2ss(b1,a1)
k=[k1a k1b; k1c k1d]'
I=[1; 1]
ac=ga-gb*k*I*gc
```

```
p=length(ac)
G=kron(eye(p),ac)-kron(s.',eye(p))
V=G(1:p,1:p)
Tt=inv(V);
T=Tt'
vin=Tt*V
%Comparision of pole placement when e=0.1
af1=[-1/5 1/5 0 0; 0 -1/2 1/2 0; 0 0 0 10; 0 0 -10 -20]
bf1=[0; 0; 0; 10]
cf1=[1 0 0 0]
df1=0
eig(af1)
% Implementing controller of reduced-order model to the full-order system
% with e=0.1
Kp=[k 0 0]
ap1=af1-bf1*Kp
% Verifying the pole placement after design using reduced-order model with
% e=0.1
eig(ap1)
% Design of controller for full-order system with e=0.1
o1=[-0.3 -0.4 -10 -10.1]
Ko1=place(af1,bf1,o1)
ao1=af1-bf1*Ko1
eig(ao1)
%Comparision of pole placement when e=0.01
af2=[-1/5 1/5 0 0; 0 -1/2 1/2 0; 0 0 0 100; 0 0 -100 -200]
bf2=[0; 0; 0; 100]
cf2=[1 0 0 0]
df2=0
eig(af2)
% Implementing controller of reduced-order model to the full-order system
% with e=0.1
Kp=[k 0 0]
ap2=af2-bf2*Kp
% Verifying the pole placement after design using reduced-order model with
% e=0.1
eig(ap2)
% Design of controller for full-order system with e=0.01
o2=[-0.3 -0.4 -100 -101]
Ko2=place(af2,bf2,o2)
ao2=af2-bf2*Ko2
eig(ao2)
% Verifying the pole placement after design using full-order model with
% e=0.1
eig(ao1)
eig(ap1)
% Verifying the pole placement after design using full-order model with
% e=0.01
eig(ao2)
eig(ap2)
```

APPENDIX D

MATLAB M-File for Decentralized Controller of Full-Order Power System

```
% Full-order system

af=[-0.5 0 0.1 -0.2 0.1; 0 -1 0.1 0.3 -0.2; 4 3 -5 0 0; -4 4 0 -4.5 0; 3.5 3
0 0 -4]
bf=[1 0 ; 0 1 ; 4 0 ; 0 -5 ; 0 6]
eig(af)

a11=af(1:2,1:2)
a12=af(1:2,3:5)
a21=af(3:5,1:2)
a22=af(3:5,3:5)
b1=bf(1:2,1:2)
b2=bf(3:5,1:2)
eig(a11)
eig(a22)
p1=[-1 -0.5]
p2=[-4 -4.5 -5]
k1=place(a11,b1,p1)
k2=place(a22,b2,p2)
a1=a11-b1*k1
a2=a22-b2*k2

eig(a1)
[V1, D1] = eig(a1)
eig(a2)
[V2, D2] = eig(a2)
M1=[real(V1(1,1)) real(V1(1,2)); real(V1(2,1)) real(V1(2,2))]
M2=[real(V2(1,1)) real(V2(1,2)) real(V2(1,3)); real(V2(2,1)) real(V2(2,2))
real(V2(2,3)); real(V2(3,1)) real(V2(3,2)) real(V2(3,3))]

af11=(inv(M1))*a1*M1
af22=(inv(M2))*a2*M2
af12=(inv(M1))*a12*M2
af21=(inv(M2))*a21*M1
bf1=(inv(M1))*b1
bf2=(inv(M2))*b2

k12=((inv(bf1'*bf1))*bf1'*af12)'
k21=((inv(bf2'*bf2))*bf2'*af21)'

AF11=af11
AF22=af22
AF12=(eye(2)-bf1*(inv(bf1'*bf1))*bf1')*af12
AF21=(eye(3)-bf2*(inv(bf2'*bf2))*bf2')*af21

A=[AF11 AF12; AF21 AF22]
eig(A)
```

APPENDIX E

MATLAB M-File for Reduced-Order Decentralized Controller Design for State Feedback of a Decentralized Power System

```
% Full-order system

af=[-0.5 0 0.1 -0.2 0.1; 0 -1 0.1 0.3 -0.2; 4 3 -5 0 0; -4 4 0 -4.5 0; 3.5 3
0 0 -4]
bf=[1 0 ; 0 1 ; 4 0 ; 0 -5 ; 0 6]
eig(af)

a11=af(1:2,1:2)
a12=af(1:2,3:5)
a21=af(3:5,1:2)
a22=af(3:5,3:5)
b1=bf(1:2,1:2)
b2=bf(3:5,1:2)
eig(a11)
eig(a22)
p1=[-1 -0.5]
p2=[-4 -4.5 -5]

k1=place(a11,b1,p1)
k2=place(a22,b2,p2)
a1=a11-b1*k1
a2=a22-b2*k2
eig(a1)
[V1, D1] = eig(a1)
eig(a2)
[V2, D2] = eig(a2)
M1=[real(V1(1,1)) real(V1(1,2)); real(V1(2,1)) real(V1(2,2))]
M2=[real(V2(1,1)) real(V2(1,2)) real(V2(1,3)); real(V2(2,1)) real(V2(2,2))
real(V2(2,3)); real(V2(3,1)) real(V2(3,2)) real(V2(3,3))]

af11=(inv(M1))*a1*M1
af22=(inv(M2))*a2*M2
af12=(inv(M1))*a12*M2
af21=(inv(M2))*a21*M1
bf1=(inv(M1))*b1
bf2=(inv(M2))*b2

k12=((inv(bf1'*bf1))*bf1'*af12)'
k21=((inv(bf2'*bf2))*bf2'*af21)'

AF11=af11
AF22=af22
AF12=(eye(2)-bf1*(inv(bf1'*bf1))*bf1')*af12
AF21=(eye(3)-bf2*(inv(bf2'*bf2))*bf2')*af21

A=[AF11 AF12; AF21 AF22]

eig(A)
```