University of Southern Queensland

Faculty Of Engineering and Surveying

Functional Observer For Glucose Regulation

A dissertation submitted by

Marina Shanil Mendis

In fulfillment of the requirement of

Courses ENG4111 and 4112 Research Project

Towards the degree of

Bachelor of Electronic and Computer Systems Engineering

Submitted: October, 2008

University of Southern Queensland
Faculty of Engineering and Surveying

ENG4111 & ENG4112 Research Project

Limitations of Use

Certification

I certify that the ideas, design and experimental work, results and, analyses and conclusions set out in this dissertation are entirely my own effort, except where otherwise indicated and acknowledged.

I further certify that the work is original and has not been previously submitted for assessment in any other course or institution, except where specifically stated.

Marina Shanil Mendis

Student Number: 0050029217

....
Signature

....
Date

Abstract

Ever since David Luenberger introduced the problem of observing the state vector of a deterministic linear time-invariant multi-variable system [1] [2], there have been numerous studies on the design and optimisation of observers for reduced-order. Of particular interest is one type of observer presented in [1], the functional observer.

One of the problems of functional observers is the complexity in their design, proposed design methods seem to offer a trade-off between the observers' general applicability to a wide range of systems, degree of order reduction and simplicity. In this project we attempt to take a non-linear 19^{th} order insulin-glucose physiological model, linearise the system, apply a model reduction technique and design a state feedback controller which utilises Darouach's functional observer to supply the linear function of the states for feedback. A new result that relaxes the sufficient and necessary conditions for the existence of the Darouach observer is discussed. This result increases the range of systems to which this observer can be applied.

Acknowledgements

In undertaking this project, there are many people who have made contributions. I would like to take this opportunity to acknowledge their help.

First of all, I would like to thank Senior Lecturer Dr Paul Wen for giving me the opportunity to take on this project. He has been a fantastic supervisor, his enthusiasm and feedback throughout the year is greatly appreciated. Thank you for taking time away from your very busy schedule to discuss and improve my project.

A special thanks to my husband Sohan for his ongoing support, tolerance and looking after our three beautiful children giving me the opportunity to achieve my dreams. For my children thanks for letting mummy be both mother and student – thank you also for your patience. I would also like to thank my parents Stella and Lionel, for their words of encouragement and for just being around.

During my tenure at university I have had the pleasure of studying with some great company, thank you for making university so vibrant and memorable place.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Control systems are used to regulate the behaviour of dynamic systems and are found in a variety of machines, products and processes. They control quantities such as motion, speed, heat flow, fluid pressure, tension, voltage, and current. Many control theory concepts are based on sensors measuring quantities that describe the state of the system. The desired output can then be achieved by manipulating the inputs of the system. This common control scheme where the system state vector is made available to the controller is known as feedback control. In some instances physical sensors can degrade the performance of a control system. According to Ellis [4], physical sensors can suffer from numerous common problems. First, sensors and their associated cabling can be expensive, raising the overall cost of the control system. Second, it can be impractical to measure some signals for reasons such as a harsh operating environment, relative motion between the sensor and the controller or the object of interest not being readily accessible. Third, sensors introduce inherent errors such as stochastic noise, cyclical errors, and bounded responsiveness. These limitations prevent all the state variables necessary to describe the state of the system from being provided. Hence only an incomplete state vector can be supplied for feedback. This very issue has led to research into the state observer which would estimate those state variables.

1.2 Observers

The state observation problem centres on the reconstuction of all the state variables using knowledge of available system inputs and outputs. Known as an observer or state reconstructor, observers can be used to augment or replace sensors altogether in a control system. Observers are algorithms that combine sensed signals with knowledge about the control system to produce estimated or observed signals. These estimated signals can be more accurate, less expensive to produce and more reliable than sensed signals. Observer theory offers designers a powerful alternative to having sensors measure quantities for the control system [4]. Significant developments in observer literature dates back to the 1960s through the research of Luenberger and Kalman. David Luenberger's papers have prompted further research into the design, simplification and optimisation of observers. His paper "Observers for Multivariate Systems" in 1966 presents three distinct types of observers: the Full-State Observer, Reduced-Order Observer and the Functional Observer. Of particular interest is the fact that Luenberger mentions the possibility of reconstructing a single linear functional of the unknown state vector [1]. It is frequently the case that only a linear function of the system state vector is required for feedback; as Luenberger notes [1] this situation arises in the design of a linear, time-invariant, state feedback for a single-input system. Typically a linear feedback control law of the form Kx is required to be estimated; this signal can be provided by the functional observer.

1.3 Linear State Space Systems

In order to design a control system, a mathematical model that describes the behaviour of the system must first be produced. There are various ways of achieving this, with state space representation being one possible method of modelling the physical system. State space representation models the physical system as a set of input, output and state variables related by first-order differential equations all expressed in matrix form.

The dynamic behaviour of many systems can at any time be described by the linear time-invariant model

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{1.1}$$

$$y(t) = Cx(t) \tag{1.2}$$

where $x(t) \in \mathbb{R}^n$ is the system state, $u(t) \in \mathbb{R}^p$ is the control input to the system, and the output of the system $y(t) \in \mathbb{R}^m$ represents some linear combination of the system states $x(t)$. The state of a dynamic system is the smallest set of (state) variables such that the knowledge of these variables at time $t = t_0$ and knowledge of the input for $t \geq 0$ is sufficient to determine the system for $t \geq t_0$ [5]. The matrices A,B and C are assumed to have compatible dimensions and for time-invariant systems are independent of time (constant). The inputs and outputs of the system are expressed as a function of the system state and time, so that for any given state, the input can be determined to drive a particular output.

1.4 Project Motivation

The need for functional observers stems from the ubiquitous nature of state feedback control strategies. Modern optimal control laws almost invariably have control commands based ultimately upon the state variables. In addition, useful design tools such as pole placement are most easily implemented in a state feedback framework. As previously mentioned, a complete state vector is rarely available for use in state feedback, so it is necessary for the design to be modified. A useful approach is to use state estimates in place of actual state variables for use in the control law, these estimates will come from an observer or Kalman filter. In order to limit the scope of the research, the discussions will be limited to the observer.

Luenberger, in his 1966 paper [1], introduces three types of observers, two of which have been developed sufficiently to be used in industry. The general problem of reconstructing linear functions of the state is still an open problem due to the great complexities of the problem. However, over the forty odd years that has past since it was first raised, significant progress has been made in the literature. The attractive features of the functional observer however, are undeniable. It's major advantage is that it is of much lower order than can be achieved with a full-order or reduced order observer. A lower order reduces computational burden and hence provides a relatively economical option to the designer. The functional observer also combines the observer and controller into a single entity which removes the need to place observer poles. Furthermore the single unit reduces wiring, eliminating a source of cabling and connector noise.

1.5 Review of Literature

To implement a state-variable feedback control law for a system in which the states are not measurable directly, the values of state variable, or the value of a linear function of the states must be estimated, using historical and current data for both the input and output variables. A solution to this problem was presented by Kalman [6], [7] who proposed the so-called Kalman Filter for use in optimal feedback control schemes with a high signal to noise ratio [8]. This solution is not appropriate in the general case, being too heavily influenced by disturbances and by placing additional constraints on the system. Kalman filters are also, by definition, of non-minimal order, requiring that the filter order match the order of the system. In 1964 Luenberger [9] [1], showed that an *observer* can be constructed for the continuous time deterministic case which produces stable results with errors tending to zero as time approaches infinity. The so-called Luenberger filter also reduces the necessary order of the observer to $n - p$ where n is the system order and p is the number of outputs. Luenberger's earlier work was expanded upon significantly in the next 7 years to include time-varying systems, discrete systems and stochastic systems [10], [11] and [12]. In 1971 Luenberger released a paper [2] summarising the results of the accumulated works.

The construction of an observer with a lower dimension than a full state observer was first presented by Bass and Gura [13] and Luenberger [9] [1]. In many applications a state observer, one which observes all state variables, can not be designed due to restrictive design conditions. Details of these restrictions are discussed later and are presented in many papers, [1], [2], [14], and [15]. In the case of a large system, the observer order required to estimate the states is unreasonably high, making implementation impractical. Fortunately, estimation of all states is rarely required. For the case where a linear combination of two or more states is sufficient, a *functional observer* can be designed under weaker conditions [16] and of a far lower order [1].

The development of functional observers can be mapped by a number of key papers, though this is by no means intended as an exhaustive list of techniques. Rather, it serves to illuminate the direction which the development of functional observers has taken.

1.5.1 Fortmann and Williamson - 1972

Fortmann and Williamson [17] is one of the first papers to address necessary and sufficient conditions for the existence of a functional observer. An algorithm for determining a minimal order q single output observer of given output and pole configuration is outlined. Fortmann and Williamson also supply an algorithm where the multi-output case is reduced via a canonical form to an output-coupled set of single output systems, which can then be treated by a single output approach. Finally, they developed a method which, unlike Luenberger's, does not require the observer matrix and the system matrix A have different poles.

1.5.2 Murdoch - 1973

Murdoch [18] notes that the design algorithm presented by Luenberger is impractical when applied to large systems and in this paper Murdoch presents an algorithm more suitable for use on a digital computer than the previous methods. The paper presents a design of an observer of minimum order and arbitrary dynamics that provides a linear function of the state vector. Murdoch states that minimum order of the observer is $(v-1)$ where v is the observability index of the system. Murdoch's method gives us the freedom to arbitrarily assign observer eigenvalues with the same mild conditions of Luenberger's, that the eigenvalues of N be distinct from those of A

1.5.3 Moore and Ledwich - 1975

Moore and Ledwich [19] seeks to understand the necessary and sufficient conditions surrounding the design and existence of a pth order observer. The technique is lengthy and based on decision theory, the mathematics is arduous and the authors do not use standard notation.

1.5.4 Gupta, Fairman and Hinamoto - 1981

Gupta *et al* [8] presented a method for the design of an observer for the reconstruction of a single linear functional of the state variables for a finite-dimensional system. The aim of the method is to construct an observer (which does not require the system to be transformed to a Luenberger canonical form which would place restrictions upon the selection of eigenvalues in the system). It has been observed [20] and [21] that while the observer eigenvalues need only lie in the left hand side of the complex plane to ensure stability and asymptotic estimation, the eigenspectrum of the observer must be determined by additional requirements in certain applications.

1.5.5 Tsui - 1985

Tsui's paper [15] presents some modifications to the Gupta *et al.* algorithm. The most significant improvement being the use of a transformation to place A in lower block Hessenberg form. The Hessenberg form is a triangular matrix with additional elements on the sub-diagonal. The matrix possesses some useful properties for these applications; it has an easily calculated rank and provides the observability index by definition. Tsui's algorithm modifies some of the computation which, whilst not significantly reducing the number of calculations to be made, reduces the error in the calculation. Although this is rather insignificant given the speed and accuracy of today's computers, the method does play a part in reducing the error in a feedback system. Again, this particular solution is suitable for a time-invariant linear system requiring a single linear feedback.

In the same year Tsui also published a paper on the design of Multifunctional Observers [22]. This algorithm still allows for arbitrary eigenvalues, to ensure stability, and a maximum order of $m(v-1)$ where m is the number of functions which are required and v is the observability index of (A, C). The order of the system is, however, significantly reduced for functionals which share components. This paper uses a similar method to the single functional algorithm relying on a similar transformation into lower Hessenberg form. This method has a wider range of applications than single functional observers. For instance, observers calculated in this manner could be used to control more than one system with the same state variables. It can also handle multiple input systems or systems which have different feedback systems depending on some external variables.

1.5.6 Aldeen and Trinh - 1999

Aldeen and Trinh [23] introduces a design method for the functional observer, which bases the observer order on the ratio of independent output measurements to independent input measurements. For high order systems which have far more outputs than inputs, the order can be reduced to the minimum proposed by Murdoch [24] and Tsui [15]. This observer design method is relatively easy to implement and only requires the linear equations be satisfied along with typical observer constraints. An advantage of this method over those proposed by Tsui [22] [25] and Gupta *et. al.* [8] is that the system matrices do not need to be transformed to any other canonical form. The obvious drawback to this method is that systems with a considerably large number of inputs compared with the system outputs would have a large order approaching (or equal to) that required in a full-state observer.

1.5.7 Datta and Sarkissian - 2000

Datta and Sarkissian [26] introduces a new algorithm designed for high performance computing to solve the Syvester-observer equations. The paper also includes a new algorithm for the direct, and efficient, implementation of the standard functional observer. Both algorithms are compared to Van Dooren's algorithm for reduced-order state estimation and are demonstrated to perform better than Van Dooren's algorithm [27].

1.5.8 Darouach - 2000

Aldeen and Trinh [23] provides a simple algorithm for the design of a functional observer. The algorithm provides sufficient conditions for the existence of such an observer, but does not provide the necessary conditions. This implies where the algorithm fails to yield a result (non-existence), nothing more is known about the system. It is desirable for the algorithm to include necessary conditions for its use to accompany the sufficient conditions of its design. Darouach presents the necessary and sufficient conditions for the existence of an r^{th} order observer, where r is the required number of functions to be estimated. Darouach [16] provides the necessary and sufficient conditions for both the continuous-time and discrete-time systems and as well as an elegant algorithm for designing the observer.

1.6 Contributions to the Dissertation

This dissertation has been written as an introductory reference for functional observers. It is expected that the reader has a strong grasp of fundamental linear algebra and calculus. To facilitate understanding of the concepts discussed, I have attempted to provide annotated derivations for many of the key results within the literature. If the reader requires a more rigorous handling of the concepts, I have endeavoured to provide relevant citation to journal papers for further reading.

The application chosen to demonstrate the order-reducing ability of the functional observer is in the field of glucose regulation. The human glucose-insulin model used in this study is of suitably high order. In implementing the functional observer we encounter the many issues related to system minimalisation and controller design.

1.7 Summary

We have now defined the topic of functional observers and gained some insight into the work that has been done in this area of research. It is important to note that the problem of solving for a minimal order functional observer that is applicable to wide class of systems is still an open one and this study will illustrate some of those limitations in one of the current techniques.

1.8 Dissertation Structure

This dissertation comprises of the following chapters:

- Chapter 1 (this chapter) introduces the problem and deals with the main motivation behind the project. A literature review of various approaches to solving the observer problem is presented.
- Chapter 2 provides a background to some of the mathematical and theoretical concepts that are required in the analysis and eventual implementation of the functional observer.
- Chapter 3 introduces the full-state observer and the reduced-order observer and describes the construction of each.
- Chapter 4 builds upon the knowledge of observers from Chapter 3 which culminates in the functional observer problem. Standard notation is defined for the detailed analysis of Darouach's functional observer.
- Chapter 5 provides an application for the functional observer. A glucose-insulin regulation problem using pole-placement is presented. A lengthy design process that begins from a 19^{th} order non-linear system is provided. The design process entails linearisation, model reduction through balanced truncation technique, control law design and finally a functional observer implementation to provide the linear functional state feedback.
- Chapter 6 concludes the paper, summarising the results and provides some suggestions to the reader for further areas of research.
- Appendix A documents the project specification.
- Appendix B contains a collection of Matlab code that was written to obtain the results for Chapter 5.

Chapter 2

Background

The following sections introduce some of the key concepts and tools that are used in the literature.

2.1 Linearisation

One may question why most control theory results apply to linear systems when in fact most physical processes are non-linear in nature. There are at least two reasons for concentrating on linear systems:

- There is extensive literature on and knowledge of vectors and matrices from linear algebra, and they are powerful tools in the analysis of linear systems. Generalising a system to the non-linear case seldom leads to results that are amenable to implementation except in special cases.
- The linear model can in many cases provide reasonable approximations to the non-linear model, because of the fact that the system is being controlled. The linear assumption becomes increasingly valid as the system operates closer to some operating point from which the linear approximation is taken. The very fact that control systems tend to regulate a system close to some operating point ensures that the linearisation approximation is reasonable.

As Khalil [28] points out, due to the powerful tools available for linear systems we should, whenever possible, make use of linearisation to learn as much as we can about the behaviour of a non-linear system. Let us now examine how a linear model can be used to form an approximation of a non-linear system.

A non-linear model is typically of the form

$$\dot{x}(t) = f[x(t), u(t), t] \tag{2.1}$$

$$y(t) = g[x(t), t] \tag{2.2}$$

A linear model of the form found in (1.1) and (1.2) can be used as an approximation to the non-linear system presented in (2.1) and (2.2) by *linearising* the non-linear system about some nominal state trajectory $x_0(t)$ and a nominal input $u_0(t)$. If we consider small excursions from the nominal value, that is $\delta x(t) \triangleq x(t) - x_0(t)$ and $\delta u(t) \triangleq u(t) - u_0(t)$, a Taylor series expansion of (2.1) and (2.2) about $[x_0(t), u_0(t), t]$ will yield

$$f[x(t), u(t), t] = f[x_0(t), u_0(t), t] + A_0(t)\delta x(t) + B_0(t)\delta u(t) + \alpha_0[x(t), u(t), t] \quad (2.3)$$

$$g[x(t), t] = g[x_0(t), t] + C_0(t)\delta x(t) + \beta_0[\delta x(t), t] \quad (2.4)$$

where $\alpha_0[x(t), u(t), t]$ and $\beta_0[\delta x(t), t]$ are the higher-order terms (second and higher) in the Taylor series expansion. The matrices

$$A_0(t) \triangleq \left.\frac{\partial f}{\partial x}\right|_{x_0,u_0,t} \quad , \quad B_0(t) \triangleq \left.\frac{\partial f}{\partial u}\right|_{x_0,u_0,t}$$
$$C_0(t) \triangleq \left.\frac{\partial g}{\partial x}\right|_{x_0,u_0,t} \quad (2.5)$$

are Jacobian matrices of appropriate dimensions (see definition in Section 4.1), evaluated at known system nominal values $[x_0(t), u_0(t), t]$. By ignoring the second and higher-order expansion terms, we effectively *linearise* the system. This yields the following

$$\delta\dot{x}(t) = A_0(t)\delta x(t) + B_0\delta u(t) \quad (2.6)$$

$$\delta y(t) = C_0(t)\delta x(t) \quad (2.7)$$

Both equations (2.6) and (2.7) form the linearised perturbation model. This model provides a close approximation of the non-linear model provided that the non-linear effects are relatively small, or equivalently, provided that higher-order expansion terms $\alpha_0[x(t), u(t), t]$ and $\beta_0[\delta x(t), t]$ are small for all time t.

This relationship is shown graphically in Figure 2.1 where a new set of axes, δx and $\delta f(x)$, is created about point A, and $f(x)$ is approximately equal to $f(x_0)$ for small perturbations about this new origin.

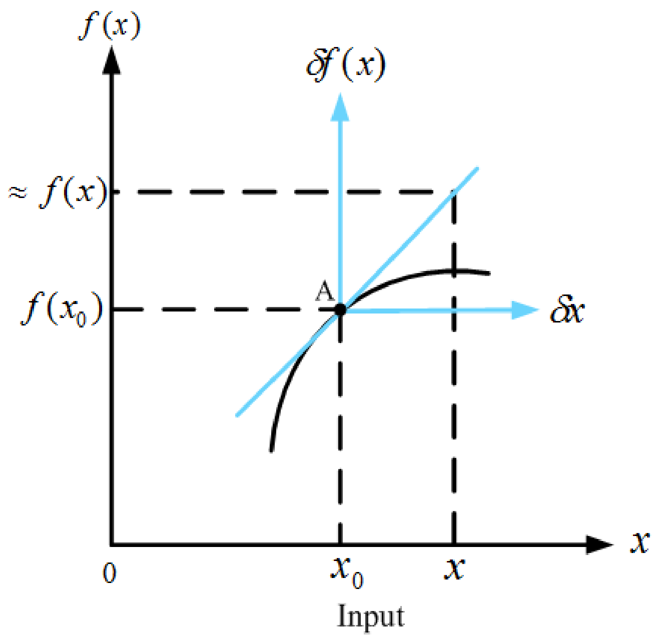

Figure 2.1: Non-linear System Linearised about Point A

2.2 Controllability and Observability

There are two properties of linear state-space descriptions that are often necessary assumptions about a system for controller and observer construction. In order to construct an observer we require that the system be *minimal*, that is the system must be controllable and observable. Essentially these terms imply that for a system of equations it is possible to control the state of each variable by altering the inputs (controllable), and that each state variable can be extracted by measuring the output of the system. The following definitions appear in O'Reilly [14].

Definition 1.1 [14] pg.6
The state of the continuous-time linear system (1.1), (1.2) is said to be *reachable (from the zero state)* at time t if there exists a $\tau \leq t$ and an input $u \in R^r$ which transfers the zero state at time τ to the state x at time t.

Definition 1.2 [14] pg.6
The continuous-time linear system (1.1), (1.2) is said to be *controllable (to the zero state)* if, given any initial state $x(\tau)$, there exists a $t \geq \tau$ and a $u \in R^r$ such that $x(t) = 0$.

Definition 1.3 [14] pg.7
Let $y(t; \tau, x, u)$ denote the output response of the linear system (1.1), (1.2) to the initial state $x(\tau)$. Then the (present) state $x(\tau)$ of the linear system is *unobservable* if the (future)

output

$$y(t; \tau, x, 0) = 0$$

for all $t \geq \tau$.

The proofs accompanying these definitions are given by Kalman *et al.* [29].

A useful example is to consider a diagonal system, as it is in parallel form.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} a_1 & 0 & & 0 \\ 0 & a_2 & & 0 \\ & & \ddots & \\ 0 & 0 & & a_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} u \tag{2.8}$$

$$y = \begin{bmatrix} c_1 & c_2 & \dots & c_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \tag{2.9}$$

This system realisation will be controllable if and only if $b_i \neq 0$ for $i = 1, 2, \dots, n$ and the system is completely observable if and only if $c_i \neq 0$ for $i = 1, 2, \dots, n$. Although this is a simple example it succinctly demonstrates the potential difficulties. Each state variable x_i is affected by the input u scaled by a respective value of b_i, if any b_i is zero then it follows that the input cannot affect the variable x_i. Similarly if the respective value of c_i is zero then the variable does not make any contribution to the output y, and so cannot be extracted from the output data. This becomes more difficult to note as the system grows more complex, because the input may affect variables which themselves form part of the equations for other state variables (these are indirectly affected by the inputs). Similarly if the output y is composed of only some variables, but these are in turn made from other variables, then the secondary variables are also observable through the system output.

To overcome these intuitive difficulties in determining whether a system is either controllable or observable, Controllability and Observability matrices are defined. For the standard system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \end{aligned} \tag{2.10}$$

we define the Controllability matrix as

$$\mathcal{C} = \begin{bmatrix} B & AB & \cdots & A^{n-1}B \end{bmatrix} \tag{2.11}$$

for an n^{th} order system. It can be shown [30] that for a system to be completely controllable

$$\text{rank}\,[\mathcal{C}] = n \tag{2.12}$$

The controllability matrix should contain n linearly independent columns (or rows). Since the test applies to the matrices, it is sometimes said that the matrix pair (A, B) is controllable if (2.12) holds.

Similarly, we define the Observability matrix $\mathcal{O}$ as

$$\mathcal{O} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix} \tag{2.13}$$

Again, it is a requirement that $\mathcal{O}$ contains n linearly independant rows (or columns) for system observability, which is equivalent to

$$\text{rank}[\mathcal{O}] = n \tag{2.14}$$

When a system is both controllable and observable we refer to it as a *minimal* system. Luenberger indicates that for an observer to exist the system must be minimal. This is broadly correct for a full state Luenberger observer. Our focus, however, is on Functional Observers. Although it can be shown [14] that those state variables which are included in the feedback must necessarily be observable, it is not necessary that the entire system be minimal. This is of great significance when reducing the complexity and order of functional observers, and will become apparent in the following sections.

2.2.1 Controllability and Observability Index

The controllability and observability indices are found by reducing the matrices to find linearly independent rows and by allocating a number beginning from zero to each independent row. This idea was introduced by Luenberger [1] and is used in many papers as the lower bound on observer order. O'Reilly [14] showed that for a full system observer the minimum order is $v-1$, where v is the observability index. We find that this condition can be relaxed for the functional observer case and indeed it is the lower bound only when we wish to arbitrarily allocate the system eigenvalues. The indices are defined as follows, for a system that is controllable/(observable), there exists a smallest number $k/(v)$ such that

$$rank \left[\begin{array}{ccccccccc} B & \vdots & AB & \vdots & A^2B & \vdots & \cdots & \vdots & A^{k-1}B \end{array} \right] = n \quad \text{(Controllability Index)}$$

$$rank \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{v-1} \end{bmatrix} = n \qquad \text{(Observability Index)} \tag{2.15}$$

2.3 Minimal Realisations

A system is minimal if and only if it is fully state controllable and fully observable. Minimal systems are invariant to similarity transformations, so neither controllability or observability of a minimal system is affected by similarity transformations. However, as we shall see in Chapter 5, not all the systems are minimal. It would therefore be useful to have techniques in which non-controllable and/or non-observable systems can be represented and also have methods for these non-controllable/non-observable state variables to be separated out. One possible way in which this can be achieved is via similarity transformations. Much of this discussion on realisations can be attributed to T. Kailath [31].

2.3.1 Similarity Transformations

The state-space equations used to model a system are non-unique. Therefore it is possible to transform a given system into another system of a different form whilst still maintaining the same input/output behaviour.

A non-singular transformation matrix T_1 can be chosen to transform the state vector as shown

$$x(t) = T_1\hat{x}(t) \tag{2.16}$$

By taking the derivative of (2.16) and substituting into the state-space equations (2.10) we get

$$\begin{aligned} \dot{\hat{x}}(t) &= (T_1^{-1}AT_1)\hat{x}(t) + (T_1^{-1}B)u(t) \\ y(t) &= (CT_1)\hat{x}(t) + (D)u(t) \end{aligned}$$

The two realisations $\{A, B, C, D\}$ and $\{T_1^{-1}AT_1, T_1^{-1}B, CT_1, D\}$ are called similar realisations. Alternatively, if the transformation matrix is defined as $\hat{x}(t) = T_2x(t)$, the similar realisation is $\{T_2AT_2^{-1}, T_2B, CT_2^{-1}, D\}$.

2.3.2 Representation of Non-controllable Realisations

Let A, B, C be such that if

$$\text{rank } \mathcal{C}(A, B) = r < n \tag{2.17}$$

Then there is always a transformation matrix T such that the realisation

$$\left\{ \bar{A} = T^{-1}AT, \quad \bar{B} = T^{-1}B, \quad \bar{C} = CT \right\} \tag{2.18}$$

has the form

$$\bar{A} = \begin{bmatrix} \bar{A}_c & \bar{A}_{c\bar{c}} \\ 0 & \bar{A}_{\bar{c}} \end{bmatrix}, \quad \bar{B} = \begin{bmatrix} \bar{B}_c \\ 0 \end{bmatrix}, \quad \bar{C} = \begin{bmatrix} \bar{C}_c & \bar{C}_{\bar{c}} \end{bmatrix} \tag{2.19}$$

where the transformation matrix T of rank n has the following structure

$$T = \begin{bmatrix} B & AB & \ldots & A^{r-1}B & v_1 & v_2 & \ldots & v_{n-r} \end{bmatrix} \tag{2.20}$$

where the last $n - r$ column vectors $v_1, v_2, \ldots, v_{n-r}$ are chosen such that they are linearly independent of the first r columns of the transformation matrix T.

This realisation has the following two useful properties,

1. The $r \times r$ subsystem $\{\bar{A}_c, \bar{B}_c, \bar{C}_c\}$ is controllable.

2. The subsystem has the same transfer function as the original system.

2.3.3 Representation of Non-Observable Realisations

A dual statement can be made about the non-observable realisation. Thus if

$$\text{rank } \mathcal{O}(C, A) = r < n \tag{2.21}$$

we can find a nonsingular matrix T such that the realisation

$$\left\{ \bar{A} = TAT^{-1}, \quad \bar{B} = TB, \quad \bar{C} = CT^{-1} \right\} \tag{2.22}$$

has the form

$$\bar{A} = \begin{bmatrix} \bar{A}_o & 0 \\ \bar{A}_{o\bar{o}} & \bar{A}_{\bar{o}} \end{bmatrix}, \quad \bar{B} = \begin{bmatrix} \bar{B}_o \\ \bar{B}_{\bar{o}} \end{bmatrix}, \quad \bar{C} = \begin{bmatrix} \bar{C}_o & 0 \end{bmatrix} \tag{2.23}$$

where the transformation matrix T of rank $= n$ has the structure

$$T = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{r-1} \\ v_1 \\ v_2 \\ \vdots \\ v_{n-r} \end{bmatrix} \tag{2.24}$$

This realisation has two useful properties

1. The $r \times r$ subsystem $\{\bar{A}_o, \bar{B}_o, \bar{C}_o\}$ is observable.

2. The subsystem has the same transfer function as the original system.

The above results suggests that it is possible to produce a minimal realisation from any given realisation.

2.4 Balanced Model Reduction

Balanced truncation is a simple and powerful method of model reduction first introduced by Moore [32]. The method takes the approach of transforming a given system into what is known as a balanced realisation, a form in which the Grammians are equal and diagonal. Once the system is in this form, the states are truncated from the system with the weakest states being truncated first. The relative dominance of the states in the system are given by the Hankel Singular values.

2.5 Linear State Feedback

Linear state feedback is a method in feedback control theory that allows all closed loop poles to be placed in a predetermined location. In contrast to frequency domain methods, linear state feedback methods can be applied to a wider class of systems than transform methods, namely multiple-input, multiple-output systems. One of the drawbacks of frequency domain methods of design, using either root locus or frequency response techniques, is that the method does not allow us to specify all poles of a system with an order higher than two [33]. The ability to freely place poles is desirable as the poles correspond directly to the eigenvalues of the system, which in turn control certain transient response characteristics such as overshoot, rise time, etc.

Let us consider an n^{th} order feedback control system whose n^{th} order closed loop characteristic equation is of the form

$$\det[sI - A] = s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0 = 0 \tag{2.25}$$

There are n coefficients a_i for $i = 0, 1, \ldots, n-1$ whose values can be adjusted to determine the system's closed loop pole location. Let us now consider the linear time-invariant model described by (1.1) and (1.2) where

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

In a state feedback arrangement, each state variable x_i is fed back to form the control input, u, through some gain k_i which can be adjusted to yield the desired closed loop pole locations. The feedback loop through the gains, k_i, is represented in Figure 2.2 by the feedback vector $-K$. This feedback gain is also known as a *control law*. The purpose of the control law is to allow us to assign a set of pole locations for the closed-loop system.

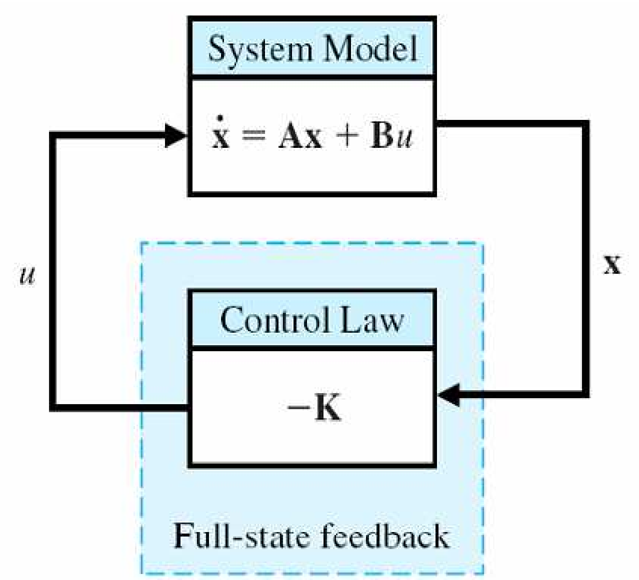

Figure 2.2: Linear-State Feedback System [3]

2.5.1 Pole Placement

Pole placement is a relatively simple state feedback design tool. Provided that the system is completely controllable, the poles of the closed loop system are arbitrarily placed in the complex plane. The ability to freely assign eigenvalues through some choice of K is generally attributed to Wonham [34], where it is assumed that complex eigenvalues only occur in conjugate pairs. Consistent with Figure 2.2,the input control law to the system is

$$u(t) = -Kx(t) \tag{2.26}$$

where

$$K = \begin{bmatrix} k_1 & k_2 & \cdots & k_n \end{bmatrix} \tag{2.27}$$

Based on our system equations (1.1), (1.2) and control law equation (2.26), the following closed loop system equations are obtained

$$\dot{x}(t) = (A - BK)x(t) \tag{2.28}$$

$$y(t) = Cx(t) \tag{2.29}$$

It can be readily shown that the closed loop characteristic equation can be written as

$$\begin{aligned} \det(sI - (A - BK)) = s^n &+ (a_{n-1} + k_n)s^{n-1} + (a_{n-2} + k_{n-1})s^{n-2} \\ &+ \cdots + (a_1 + k_2)s + (a_0 + k_1) = 0 \end{aligned} \tag{2.30}$$

If we now assume that the desired characteristic equation for appropriate pole placement is

$$s^n + d_{n-1}s^{n-1} + d_{n-2}s^{n-2} + \cdots + d_2s^2 + d_1s + d_0 = 0 \tag{2.31}$$

where the d_i's are the desired coefficients, we equate equations (2.30) and (2.31) to obtain

$$k_{i+1} = d_i - a_i \qquad i = 0, 1, 2, \ldots, n-1 \tag{2.32}$$

This is known as *characteristic equation matching*, and yields the feedback gain matrix K. It should be noted that there are other algorithms, such as Ackermann's formula and Bass Gura's formula which are relatively easier to implement electronically.

The next important question would be appropriate values for d_i. Unlike frequency domain methods where only a conjugate pair of complex poles need to be placed, n poles now need to be placed. There are several general guidelines for pole locations to ensure desirable closed loop system characteristics such as stability. When selecting pole locations, it is beneficial to design for minimal amount of control effort. The amount of control effort is related to how far the closed-loop poles are moved and is also increased by moving poles that are close to zeros. A pole placement philosophy that aims to minimise moving poles and concentrate on fixing only undesirable characteristics will avoid large increases in bandwidth or efforts to move poles, thus allowing smaller control actuators and associated components.

2.5.2 Linear Quadratic Regulator

The theory of optimal control is concerned with the mathematics of operating a system at minimum cost. One of the main results from the theory is provided by the linear quadratic regulator (LQR), a state feedback controller which minimises some cost function J. The primary motivation for the theory is to remove some of the guess work in iterating arbitrary pole locations (as achieved in Section 2.5.1), and instead choosing pole locations that provide a compromise between control effort, magnitude, and the speed of the response while still guaranteeing a stable system. Let us now introduce the algorithm with some comments behind the mathematical reasoning of the approach and also the final solution [35].

Again, let us continue our discussion with a state feedback system of control law $u(t) = -Kx(t)$, which yields the following system equation

$$\begin{aligned} \dot{x}(t) &= (A - BK)x(t) \\ y(t) &= Cx(t) \end{aligned} \tag{2.33}$$

The objective of the LQR is to provide a feedback gain matrix K that minimises some performance index or cost function

$$J = \int_0^\infty [x^T(t)Qx(t) + u^T(t)Ru(t)]dt \tag{2.34}$$

where Q and R are positive-semidefinite and positive-definite matrices respectively, for all t, thus ensuring that the cost function J we are minimising is non-negative. The cost function represents a weighted sum of the state and control signals energy, where Q and R are designed to account for the relative importance of the state and control signals energy.

Substituting $u(t) = -Kx(t)$ into (2.34) and factorising we obtain

$$J = \int_0^\infty [x^T(t)Qx(t) + x^T(t)K^TRKx(t)]dt = \int_0^\infty [x^T(t)(Q + K^TRK)x(t)]dt \tag{2.35}$$

This can be treated as a parameter optimisation problem where we set

$$x^T(t)[Q + K^TRK]x(t) = -\frac{d}{dt}[x^T(t)Px(t)] \tag{2.36}$$

If we then apply the product rule and substitute the closed loop system equation (2.33) we obtain

$$\begin{aligned} x^T(t)[Q + K^TRK]x(t) &= -\dot{x}^T(t)Px(t) - x^T(t)P\dot{x}(t) \\ &= -[((A - BK)x(t))^TPx(t) - x^T(t)P(A - BK)x(t)] \\ &= -x^T(t)[(A - BK)^TP + P(A - BK)]x(t) \end{aligned} \tag{2.37}$$

In comparing both sides of the equation, the equation holds true for all $x(t)$ if the following is true

$$(A - BK)^T P + P(A - BK) = -(Q + K^T RK) \tag{2.38}$$

Given that R is a symmetric positive-definite matrix we can express R as

$$R = T^T T \tag{2.39}$$

where T is some non-singular matrix.

Re-arranging and simplifying (2.38) gives

$$A^T P - K^T B^T P + PA - PBK + Q + K^T RK = 0 \tag{2.40}$$

Let us now set $K = R^{-1}B^T P$, and introduce two extra terms which do not affect the equality (the reason for this becomes clearer further on in the derivation)

$$\begin{aligned} A^T P + PA - PBK + Q + K^T RK - K^T B^T P + P^T BT^{-1} TK \\ + P^T BT^{-1}(T^T)^{-1} B^T P = 0 \end{aligned} \tag{2.41}$$

This can then be expressed as

$$A^T P + PA - PBR^{-1}B^T P + Q + [TK - (T^T)^{-1}B^T P]^T [TK - (T^T)^{-1}B^T P] = 0 \tag{2.42}$$

Minimising the cost function J with respect to K requires the minimisation of

$$x^T(t)[TK - (T^T)^{-1}B^T P]^T [TK - (T^T)^{-1}B^T P]x(t) \tag{2.43}$$

Since this expression is non-negative, the minimum occurs when it is zero, or when

$$TK = (T^T)^{-1}B^T P \tag{2.44}$$

and hence

$$K = T^{-1}(T^T)^{-1}B^T P = R^{-1}B^T P \tag{2.45}$$

The final result is a state feedback control law

$$u(t) = -Kx(t) = -R^{-1}B^T Px(t) \tag{2.46}$$

in which P must satisfy the following reduced Riccati equation to ensure that (2.42) holds.

$$A^T P + PA - PBR^{-1}B^T P + Q = 0 \tag{2.47}$$

Chapter 3

Types of Observers

Prior to discussing the role of observers in reconstructing an incomplete or inaccessible state vector of a linear system with an incomplete or inaccessible state vector, we should first consider a linear feedback system with a completely accessible state vector. If the state $x(t)$ of the open-loop system is completely available, a linear feedback control law is of the form

$$u(t) = -Kx(t) \tag{3.1}$$

The matrix K is designed to pass a linear combination of state variables into the system inputs for use in a control law. When the input is applied to (1.1), the closed-loop system is then,

$$\dot{x}(t) = (A - BK)x(t) \tag{3.2}$$

The dynamic response of the state vector in (3.2) may be asymptotically driven to the desired null vector if a feedback gain matrix K can be chosen, such that the matrix $A - BK$ has eigenvalues to the left-hand side of the complex plane.

A conceptual diagram of full-state feedback is shown in Figure 2.2 where the control law $u = -Kx$ is computed assuming full knowledge of the state vector $x(t)$. This system regulates itself against unknown disturbances affecting the output, by inputting the control law to hold the output steady. It can be readily modified to regulate itself around some desired reference signal for a tracking application.

3.1 Observer Categories

As demonstrated, the dynamic response of a controllable system can be altered through the implementation of a linear state feedback controller in (3.1). Such an application requires that system parameters and the current state $x(t)$ of the system be completely available. In the event that states are unavailable, a state observer is required to reconstruct or estimate the states that are required as inputs to the controller. The observer should simulate the actual system accurately, to provide an estimation of the states required for feedback. For the observer to be of any practical use, the error between the model and the real system be asymptotically zero.

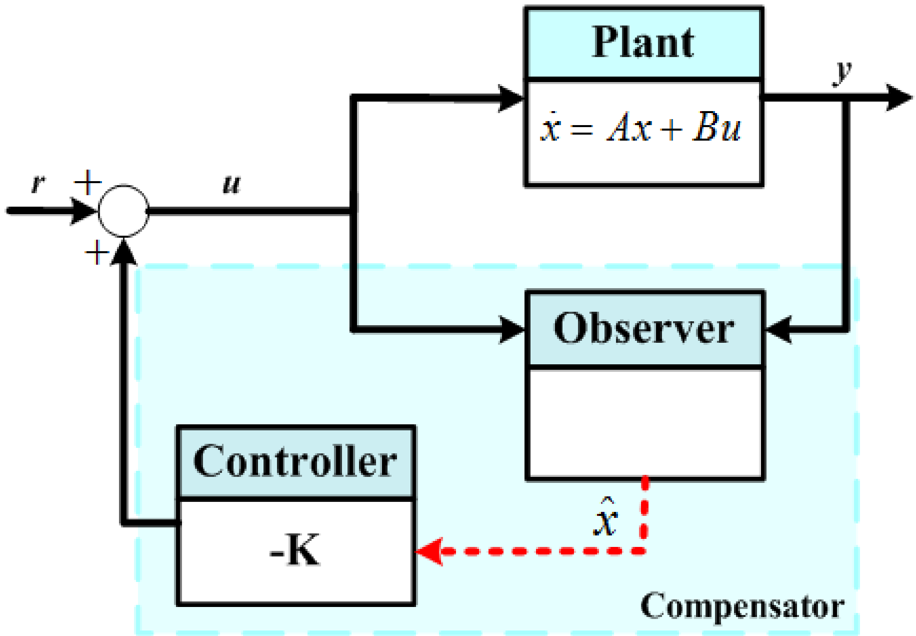

Figure 3.1: Closed-loop Observer for State Feedback

Figure 3.1 illustrates the role of the observer in reconstructing the states of the system so that it may be used for feedback into the controller. The concept of a compensator is introduced, which is simply an observer and controller working together to provide the feedback path. In this chapter the full-state observer and the reduced-order observer will be discussed in detail, to facilitate understanding of the functional observer.

3.1.1 Full-State Observer

As the name suggests, the full-state or full-order observer estimates all the states of the system. Luenberger in 1966 [1], defines this observer and introduces the conditions which must be satisfied in order for the observer to exist. It follows that a system with n state variables requires an observer of n^{th} order for full state estimation. Let us consider the linear time-invariant system of (1.1),

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{3.3}$$

$$y(t) = Cx(t) \tag{3.4}$$

For this system we now assume that $x(t)$ cannot be measured directly and that we must form an observer to reconstruct the state variables. To reconstruct all the state variables we introduce the variable $\hat{x}(t)$, denoting an approximation of the state $x(t)$ as opposed to the actual state. We define an observer with the following dynamics

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + L(y(t) - C\hat{x}(t)) \tag{3.5}$$

This form of observer uses the system inputs, $u(t)$, and outputs, $y(t)$, to estimate the state variables, $x(t)$. The first two terms are a model of the system dynamics while third term is dependent on the difference between the actual output and the expected output based on the current estimated state, and thus represents the mismatch between the system model and the actual system.

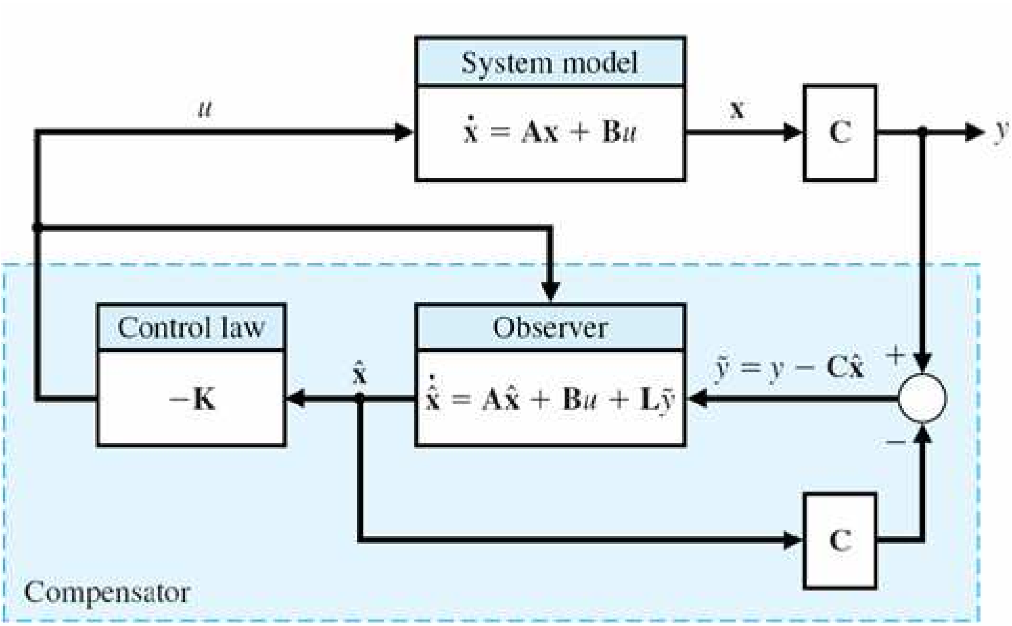

Figure 3.2: Schematic of a Full-Order Observer [3]

If we define the error to the system as

$$e(t) \triangleq x(t) - \hat{x}(t) \tag{3.6}$$

Then in taking the derivative and substituting (3.3), (3.4), and (3.5), we can obtain the error dynamics as follows

$$\begin{aligned}
\dot{e}(t) &= \dot{x}(t) - \dot{\hat{x}}(t) \\
&= (Ax(t) + Bu(t)) - (A\hat{x}(t) + Bu(t) + L(y(t) - C\hat{x}(t))) \\
&= (Ax(t) + Bu(t)) - (A\hat{x}(t) + Bu(t) + L(Cx(t) - C\hat{x}(t))) \\
&= (A - LC)x(t) - (A - LC)\hat{x}(t)
\end{aligned}$$

Incorporating the definition in (3.6), this collapses to the following differential equation,

$$\dot{e}(t) = (A - LC)e(t) \tag{3.7}$$

This differential equation can be readily solved, the solution being an exponential function of the form

$$e(t) = e^{(A-LC)t}e(t_0) \tag{3.8}$$

Clearly, for the estimation error to approach zero, the eigenvalues of exponent $(A - LC)$ must have negative real parts, or equivalently $(A - LC)$ must be Hurwitz. Since the system is time-invariant, the system parameters A and C are constant. The matrix L must therefore be chosen to satisfy those conditions via the method of pole positioning. Following the *Principle of Duality*, presented by O'Reilly [14], and based upon linear state feedback results, we shall formalise these concepts in the following Theorem

Theorem 3.1 [14] Theorem (1.16)
Corresponding to the real matrices A and C, there is a real matrix L such that the set of eigenvalues of $(A - LC)$ can be arbitrarily assigned (subject to complex eigenvalues occurring in conjugate pairs) if and only if the pair (A, C) is completely observable.

This important result defines the requirement for a full-state observer to exist. It highlights that if the original system is unobservable then it is not possible to construct a full-state observer.

Assuming however, that the system is observable, the question of how the location of the poles will affect the performance of the observer must be addressed. Since the purpose of the observer is to provide the system state to the controller, the observer must provide an estimated signal to the controller that tends toward the actual states rapidly and accurately. Otherwise the controller will yield an incorrect control signal $u = -K\hat{x}$.

The positioning of poles in $(A - LC)$ determines several performance characteristics of the observer, of which the most important is the observer transient response. As the real component of the poles become increasingly negative (move further left in the complex plane), the estimated states tend toward the actual state more rapidly. Unfortunately, this also results in the elements of L becoming larger, thus amplifying sensor noise and resulting in a less accurate signal.

We shall now consider the state feedback using the reconstructed states with the following control law

$$u(t) = -K\hat{x}(t) \tag{3.9}$$

Assuming an appropriate choice for matrix L, the estimation error $e(t)$ will be zero in steady state. The initial transient response of the full-order observer will be investigated by considering the full-order observer presented in Figure 3.2. Of particular interest when analysing the transient response, is whether the introduction of the observer will degrade system stability.

The closed loop system is completed through the interconnection of the controller (3.9) and observer (3.5) to the open-loop system (3.3), (3.4). It is described by the composite system

$$\begin{bmatrix} \dot{x}(t) \\ \dot{\hat{x}}(t) \end{bmatrix} = \begin{bmatrix} A & -BK \\ GC & A - GC - BK \end{bmatrix} \begin{bmatrix} x(t) \\ \hat{x}(t) \end{bmatrix} \tag{3.10}$$

$$y(t) = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ \hat{x}(t) \end{bmatrix} \tag{3.11}$$

From the observer definition (3.5), the definition of the reconstruction error (3.6), and prescribed control law (3.9), $\dot{x}(t)$ and $\dot{e}(t)$ can be derived as follows

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) = Ax(t) - BK\hat{x}(t) \\ &= Ax(t) - BK(x(t) - e(t)) = (A - BK)x(t) + (BK)e(t) \end{aligned} \tag{3.12}$$

$$\begin{aligned} \dot{e}(t) &= \dot{x}(t) - \dot{\hat{x}} = Ax(t) - A\hat{x}(t) + L(Cx(t) - C\hat{x}(t)) \\ &= (A - LC)e(t) \end{aligned} \tag{3.13}$$

In terms of the error between the model and real system, this results in

$$\begin{bmatrix} \dot{x}(t) \\ \dot{e}(t) \end{bmatrix} = \begin{bmatrix} A - BK & BK \\ 0 & A - LC \end{bmatrix} \begin{bmatrix} x(t) \\ e(t) \end{bmatrix} \tag{3.14}$$

$$\tag{3.15}$$

To investigate system stability, we shall consider the eigenvalues of the system by analysing the characteristic polynomial

$$\det \left(\lambda I_{2n} - \begin{bmatrix} A - BK & BK \\ 0 & A - LC \end{bmatrix} \right) \tag{3.16}$$

$$= \det\left(\lambda I_n - (A - BK)\right) \times \det\left(\lambda I_n - (A - LC)\right) \tag{3.17}$$

This leads us to a result on eigenvalue assignability and asymptotic stability of the overall closed-loop control system represented by (3.10)

Theorem 3.2 [14] Theorem (1.17)
If the linear system is completely controllable and completely observable, there exist gain matrices F and G such that the $2n$ eigenvalues of the system matrix of the closed loop

system can be arbitrarily assigned, in particular to positions in the left hand complex plane.

Note that the introduction of an observer results in an impaired transient response to the closed loop system, as a result of the non-zero reconstruction error. If designed correctly, this degradation in performance diminishes exponentially along with the error $e(t)$, thus demonstrating the application of the state reconstructor.

This form of observer is simple in its implementation. According to Theorem 3.1, as long as the system is fully observable, there are no restrictions on the pole positions of the observer. There is also no added limit on the generality of the observer as there is no requirement for the system to be given in a specific canonical form. However, the observer does suffer from the fact that it reconstructs the whole state vector, which may not necessarily be required. This inherent redundancy is addressed in the reduced-order observer.

3.1.2 Reduced-Order Observer

The full-order observer (3.5) is of order n, being equal to the number of states in the original system (1.1). Although simple both conceptually and in its construction, there are some inherent redundancies in its design. Recall our objective of reconstructing the state variables $x(t) \in \mathbb{R}^n$, and that the system outputs contain m linear combinations of the variables. Intuitively, the remaining $n-m$ states may be reconstructed by an observer of order $n - m$ which should provide the rest of the states. The following analysis will confirm this conjecture.

The design of the reduced-order observer is based on a partitioned form of the system dynamics

$$\begin{bmatrix} \dot{x}_m(t) \\ \dot{x}_u(t) \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_m(t) \\ x_u(t) \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u(t) \tag{3.18}$$

$$y(t) = \begin{bmatrix} I_m & 0 \end{bmatrix} \begin{bmatrix} x_m(t) \\ x_u(t) \end{bmatrix} \tag{3.19}$$

where x_m denotes states that are measurable and x_u denotes states that are not measurable or unknown (note that from this point, in order to assist in presenting the derivation we have dropped the (t) to indicate time-varying). The identity matrix I_m in (3.19) is an $m \times m$ matrix and the zero matrix is of dimensions $m \times (n - m)$, which gives

$$y = x_m \tag{3.20}$$

The system partition (3.18) yields the following two equations

$$\dot{x}_m = A_{11}x_m + A_{12}x_u + b_1 u \tag{3.21}$$

$$\dot{x}_u = A_{21}x_m + A_{22}x_u + b_2 u \tag{3.22}$$

The first of these equations (3.21) can be re-arranged to form an intermediate variable z

$$z = A_{12}x_b = \dot{x}_m - A_{11}x_m - B_1 u \tag{3.23}$$

The dynamics of the reduced-order observer are defined by the following

$$\dot{\hat{x}}_u = A_{22}\hat{x}_u + A_{21}x_m + B_u u + L(z - A_{12}\hat{x}_b) \tag{3.24}$$

The construction is similar to the full-order observer case, with the last term being a correction term. This correction term is based on the modelling error between the derivative of the actual measured state $\dot{x}_m$ and estimated measured state $\dot{\hat{x}}_m$.

Notice that z in (3.23) requires knowledge of $\dot{x}_m$ which may be construct by differentiating the signal. However, differentiating will severely degrade the signal if there is a small quantity of additive noise in the measurements. This difficulty can nonetheless be resolved by redefining the reduced-order estimator states in terms of a new state vector as follows

$$z = \hat{x}_u - Ly \tag{3.25}$$

Taking the derivative of (3.25) and substituting (3.20), (3.24) and (3.25) gives

$$\begin{aligned} \dot{z} &= \dot{\hat{x}}_u - L\dot{x}_m \\ &= (A_{22} - LA_{12})\hat{x}_u + (A_{21} - LA_{11})y + (B_2 - LB_1)u \end{aligned} \tag{3.26}$$

Since we require the observer to be outputting $\hat{x}_u$, $\hat{x}_u$ should not appear in the expression. We therefore make use of (3.25) to give the following

$$\begin{aligned} \dot{z} &= (A_{22} - LA_{12})(z + Ly) + (A_{21} - LA_{11})y + (B_2 - LB_1)u \\ &= (A_{22} - LA_{12})z + (A_{21} - LA_{11} + (A_{22} - LA_{12})L)y + (B_2 - LB_1)u \\ &= Fz + Gy + Hu \end{aligned} \tag{3.27}$$

where

$$F = A_{22} - LA_{12} \tag{3.28a}$$

$$G = A_{21} - LA_{11} + FL \tag{3.28b}$$

$$H = B_2 - LB_1 \tag{3.28c}$$

A schematic of the full feedback system with the reduced-order observer, represented by Equations (3.18), (3.19), and (3.27), is presented in Figure 3.3. In light of the fact that $y(t) = x_m(t)$, it is important to note how $y(t)$ acts as inputs to the dynamic part of the observer in (3.24) as well as contributing directly to the state estimate $x_u(t)$ in (3.27). This results in the estimate $x_u(t)$ being more susceptible to measurement errors in $y(t)$ than in the full-state observer case. The feedback loop is completed with the reduced-order observer providing the unknown states to augment known states. These are then made available to the controller.

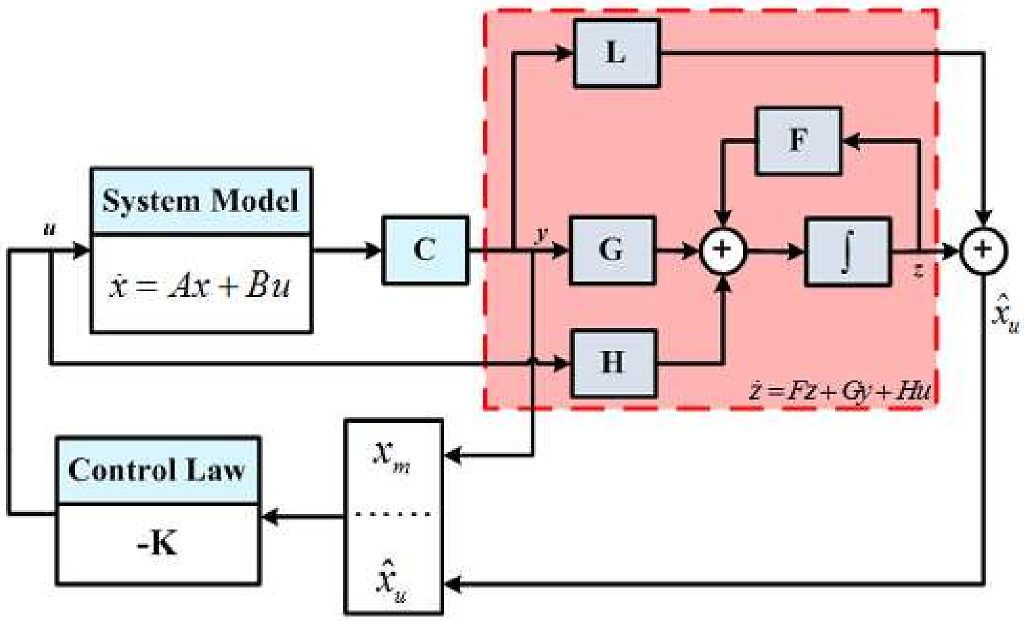

Figure 3.3: Schematic of a Reduced-Order Observer

Again, recall the error definition (3.6). Modifying this for the reduced-order case, taking the derivative to obtain (3.29), then substituting (3.22) and (3.24), followed by (3.26), we finally obtain the error dynamics in terms of the error,

$$\begin{aligned} \dot{e} &= \dot{x}_u - \dot{\hat{x}}_u && (3.29) \\ &= A_{22}(x_u - \hat{x}_u) - LA_{12}(x_u - \hat{x}_u) \\ &= (A_{22} - LA_{12})e && (3.30) \end{aligned}$$

This result is remarkably similar to that in the full-state observer case (3.7). The matrix pair (A_{12}, A_{22}) must be completely observable. A useful Lemma is presented by Luenberger [2], which states that if (C, A) is completely observable, then so is (A_{12}, A_{22}). To drive the error asymptotically to zero, the same approach as the full-state observer is taken whereby the observer gain L is chosen such that the eigenvalues of $(A_{22} - LA_{12})$ lie in the left half complex plane. Theoretically, the eigenvalues could be moved arbitrarily towards negative infinity, which would yield rapid convergence. However, this tends to result in the observer acting like a differentiator, and becoming highly sensitive to noise.

An order reduction of m is not of much significance for a single output system, especially when considering the low cost of integrated circuits. However, for multiple output systems, a somewhat more substantial reduction in observer order is possible, which reduces observer complexity.

Chapter 4

Functional Observer Design

Functional observers take advantage of a recurring theme in state feedback control. That is, frequently in feedback applications, only a linear combination or function of the state variables $Kx(t)$ is required, rather than complete knowledge of the state vector. In the previous chapter, the focus was on the reconstruction of the whole state vector, highlighting a redundant feature. The question therefore arises as to whether a less complex observer can be constructed to produce a linear function of the states. This is possible and the chapter is concerned with the construction of the functional observer, and an investigation into one particular design algorithm that has been proposed. The primary aim of the literature is to produce observers that are of further reduced order and that are stable. In addition, it is interesting to note whether the designs place any restrictions (such as pole location) on the observer which may affect its performance.

A major result for this problem was first presented by Luenberger [1], for the functional observer to be able to estimate any linear combination of states, it must itself have order of at least $v-1$ where v is known as the *observability index*. It is defined by Luenberger [2] as the least positive integer for which the matrix

$$\left[C' \vdots A'C' \vdots (A')^2C' \vdots \cdots \vdots (A')^{v-1}C'\right]$$

has rank n. This gives rise to the following theorem

Theorem 4.1 [14] Theorem (3.1)
A single linear functional of the state of a linear system can be reconstructed by an observer with $v-1$ eigenvalues that may be chosen arbitrarily (where v is the observability index of the system).

For any completely observable system $v - 1 \leq n - m$. Further, it is often the case that the order $v - 1$ of a linear functional observer is less than the order $n - m$ of the reduced-order observer. As a consequence, observing a linear function of the states may afford a significant reduction in observer order compared to observing the entire state vector.

4.1 Notation

There are numerous algorithms that have been proposed for solving the observer. One of the problems of comparing different algorithms is the choice of notation used by different authors, namely the choice of letter allocation for the system matrix parameters and their associated dimensions. For a clear and rigorous analysis of different design techniques, a consistent notation approach must be defined. To prevent confusion associated with inconsistencies, notation will be based as closely as possible to Darouach's paper [16] on functional observers, as Darouach's paper features prominently in this dissertation and as it retains much of Luenberger's [2] notation for the system parameter matrices. We can define the system as follows

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{4.1}$$

$$y(t) = Cx(t) \tag{4.2}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$, and $y(t) \in \mathbb{R}^p$. Without loss of generality, we assume that the system is completely controllable, completely observable, and that B and C are of full rank. The objective of the functional observer is to reconstruct the typical linear feedback control law of the form

$$z(t) = Lx(t) \tag{4.3}$$

where L is known and is of appropriate dimensions. In order to reconstruct the state function we require an observer of the form

$$\dot{w}(t) = Nw(t) + Jy(t) + Hu(t) \tag{4.4}$$

$$\hat{z}(t) = Dw(t) + Ey(t) \tag{4.5}$$

where $w(t) \in \mathbb{R}^q$, and $\hat{z}(t) \in \mathbb{R}^r$ is the estimate of $z(t)$. The system matrices A, B, and C and observer matrices N, J, H, D, and E are defined as follows,

$$\begin{array}{ll} A \in \mathbb{R}^{n\times n} & N \in \mathbb{R}^{q\times q} \\ B \in \mathbb{R}^{n\times m} & J \in \mathbb{R}^{q\times p} \\ C \in \mathbb{R}^{p\times n} & H \in \mathbb{R}^{q\times m} \\ L \in \mathbb{R}^{r\times n} & D \in \mathbb{R}^{r\times q} \\ & E \in \mathbb{R}^{r\times p} \end{array}$$

Figure 4.1 illustrates the role of the functional observer in reconstructing the control law to be directly fed back into the system. In contrast to the observers discussed in the previous chapter, the compensator is housed in the one observer block which, depending on the implementation, may reduce wiring and associated connection noise. The diagram also illustrates the fact that the estimated linear combination need not form part of the control strategy, namely feedback control, but could be for any purpose with the same

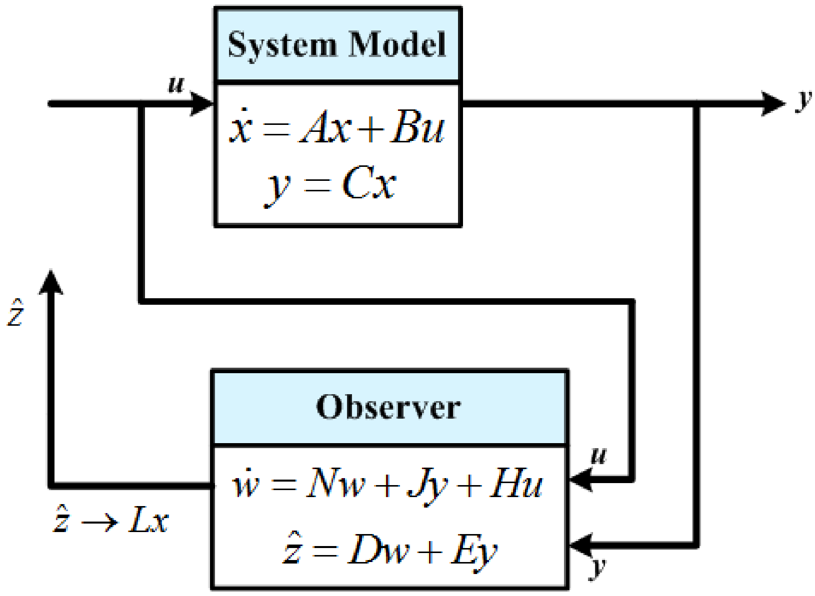

Figure 4.1: Feedback Control Law Lx implemented by a Functional Observer

design method. Also, unlike preceding chapters, in a state feedback design approach, the designed control law gain is L as opposed to K.

4.2 Functional State Reconstruction Problem

Having defined the notation to be used, we can continue with our discussion of the linear function reconstruction problem. If we assume that any states which are unobservable can be eliminated by defining a lower dimensional observer state vector, then the order q of the observer in (4.4) and (4.5) should be less than or equal to the reduced-order state observer derived in Chapter 3 ($q \leq n - m$).

The observer $w(t)$ provides a reconstruction of the state variables, while $\hat{z}(t)$ approximates $Lx(t)$. The output $\hat{z}(t)$ provides an asymptotic estimate of $Lx(t)$ if

$$\lim_{t \to \infty} [\hat{z}(t) - Lx(t)] = 0 \tag{4.6}$$

It stands to reason that if $z(t)$ estimates $Lx(t)$, then $w(t)$ estimates some other linear combination of $x(t)$, call it $Px(t)$. This is formalised by Luenberger [2] and gives rise to following theorem.

Theorem 4.2 [14] Theorem (3.2)
The completely observable q^{th} order functional observer of (4.4) *and* (4.5) *will estimate*

$Lx(t)$ if and only if the following conditions hold:

$$N \text{ is a stability matrix} \tag{4.7a}$$

$$JC = PA - NP \tag{4.7b}$$

$$H = PB \tag{4.7c}$$

$$L = DP + EC \tag{4.7d}$$

Although a formal proof of Theorem 4.2 will not be presented, it is necessary to verify and comment on the reasoning underlying the observer conditions. Taking a similar approach as the one in the previous chapter, we will define the observer error in estimating the states as

$$e(t) \triangleq w(t) - Px(t) \tag{4.8}$$

If we take the derivative of this, and then substitute the observer equation (4.4) and system equations (4.1) and (4.2) we obtain

$$\begin{aligned} \dot{e}(t) &= \dot{w}(t) - P\dot{x}(t) \\ &= Nw(t) + JCx(t) + Hu(t) - PAx(t) - PBu(t) \end{aligned} \tag{4.9}$$

Applying conditions (4.7b) and (4.7c) yields

$$\begin{aligned} \dot{e}(t) &= Nw(t) + (PA - NP)x(t) + PBu(t) - PAx(t) - PBu(t) \\ &= Nw(t) - NPx(t) \\ &= Ne(t) \end{aligned} \tag{4.10}$$

The solution to this differential equation is an exponential function of the form,

$$e(t) = e^{Nt} \tag{4.11}$$

Clearly N controls the dynamics of the observer. Applying condition (4.7a) we obtain the following result,

$$\lim_{t \to \infty} e(t) = w(t) - Px(t) = 0 \tag{4.12}$$

Our main objective is to verify that the observer correctly estimates the control law $Lx(t)$ which is the equivalent to (4.6). From the substitution of the observer equation (4.5), the substitution of the system output (4.2), and finally the application of condition (4.7d), we arrive at the following

$$\begin{aligned} e_z(t) &= \hat{z}(t) - Lx(t) \\ &= Dw(t) + ECx(t) - Lx(t) \\ &= D(w(t) - Px(t)) \end{aligned} \tag{4.13}$$

We expect this to asymptotically approach zero as a result of (4.12).

From this analysis, it is clear that the first three conditions (4.7a), (4.7b), and (4.7c) are to provide some estimate of the linear combination of the states $Px(t)$, such that its the estimates error will be asymptotically zero. Further, in the case of (4.7b), A and N are designed such that they do not share common eigenvalues, ensuring that P has a unique solution. The last condition (4.7d), then takes those states to provide the estimate for $Lx(t)$ which is verified to be convergent.

If we consider a closed loop feedback system with input $u(t) = Lx(t)$, which is estimated by the observer (4.5) in Figure 4.1, we can derive $\dot{x}$ and $\dot{e}$ by using condition (4.7d) as follows

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) = Ax(t) - B(Dw(t) + Ey(t)) \\ &= Ax(t) + BDw(t) + BECx(t) \\ &= Ax(t) + BDw(t) + B(L - DP)x(t) \\ &= (A + BL)x(t) + (BD)e(t) \end{aligned} \tag{4.14}$$

$$\dot{e}(t) = Ne(t) \tag{4.15}$$

This yields a composite system similar to that of the full-state observer in (3.14)

$$\begin{bmatrix} \dot{x}(t) \\ \dot{e}(t) \end{bmatrix} = \begin{bmatrix} A + BL & BD \\ 0 & N \end{bmatrix} \begin{bmatrix} x(t) \\ e(t) \end{bmatrix} \tag{4.16}$$

Aside from the difference in notation and the fact that the control law is $Lx(t)$ instead of $-Kx(t)$, they are very similar. Matrix element N is required to be a stability to matrix just as $A - LC$ in the full-state observer in (3.14).

Considering the constraints presented in Theorem 4.2, the functional state reconstruction problem revolves around designing the smallest possible r^{th} order observer with all observer matrices satisfying the conditions (a)-(d). Ideally, a small r should not compromise other favourable aspects of the observer, such as freedom of eigenvalue assignment and algorithm simplicity, so that it can be readily implemented in practice.

4.3 Darouach Functional Observer

EXISTENCE AND DESIGN OF FUNCTIONAL OBSERVERS
FOR LINEAR SYSTEMS: 2000 [16]

Darouach provides a simple method for designing a r^{th} order functional observer. He mentions that the solution to the functional observer problem is related to the optimal unbiased functional filter problem, especially in satisfying condition (4.7b). Darouach provides the necessary and sufficient conditions for the existence and stability of the functional observer. This culminates in a straight forward algorithm for the design of a minimal order observer based on the system parameters. The strength of this algorithm lies in its elegant implementation, which, unlike numerous other design approaches, does not require any ad hoc design decisions. This design approach also clearly specifies the necessary and sufficient conditions for the existence of the observer.

4.3.1 Summary

To be consistent with the notation prescribed in Section 4.1, we should first note that in Darouach's method that the matrix D is an identity matrix.

Darouach's analysis commences with the observer condition (4.7b). This type of equation is known as a Sylvester equation, and its solution has been the subject of considerable research. The Sylvester equation is manipulated to an equivalent form

$$PA - NP - JC = 0 \tag{4.17}$$

$$\Rightarrow (PA - NP - JC)\begin{bmatrix} L^{+} & I - L^{+}L \end{bmatrix} = 0 \tag{4.18}$$

where L^{+} denotes the Moore-Penrose generalised inverse of matrix L. If we then apply the modified condition (4.7d), where $P = L - EC$ and where we set $K = J - NE$, we obtain the following

$$N = PAL^{+} - KCL^{+} \tag{4.19}$$

and

$$P\bar{A} = K\bar{C} \tag{4.20}$$

where

$$\bar{A} = A(I - L^{+}L) \quad \text{and} \quad \bar{C} = C(I - L^{+}L) \tag{4.21}$$

Two important conditions are presented in Section 4.3.2, the first is to test for necessary and sufficient conditions for the existence of the observer, and if it does exist, a subsequent condition ensures that the matrix N is Hurwitz and the observer is stable.

4.3.2 The Algorithm

In order to satisfy the conditions in Theorem 4.2, we present the following Lemma.

Lemma 1 [16]:

$$\operatorname{rank}\begin{bmatrix} LA \\ CA \\ C \\ L \end{bmatrix} = \operatorname{rank}\begin{bmatrix} CA \\ C \\ L \end{bmatrix} \tag{4.22}$$

$$\operatorname{rank}\begin{bmatrix} sL - LA \\ CA \\ C \end{bmatrix} = \operatorname{rank}\begin{bmatrix} CA \\ C \\ L \end{bmatrix}, s \in \mathbb{C}, \Re(s) \geq 0 \tag{4.23}$$

Whether condition (4.22) is satisfied can be determined by analysing the ranks of the matrices on the *LHS* and *RHS* of 4.22). The author shows in [16] that Condition (4.23) is equivalent to the detectability of the pair (F, G), where

$$F = LAL^+ - LA(I - L^+L)\begin{bmatrix} CA(I - L^+L) \\ C(I - L^+L) \end{bmatrix}^+ \begin{bmatrix} CAL^+ \\ CL^+ \end{bmatrix} \tag{4.24}$$

$$G = \left(I - \begin{bmatrix} CA(I - L^+L) \\ C(I - L^+L) \end{bmatrix}\begin{bmatrix} CA(I - L^+L) \\ C(I - L^+L) \end{bmatrix}^+\right)\begin{bmatrix} CAL^+ \\ CL^+ \end{bmatrix} \tag{4.25}$$

where L^+ denotes the Moore-Penrose generalized inverse of matrix L. Furthermore, if matrices J, H and E satisfy Theorem 4.2, a Hurwitz matrix N is given by

$$N = F - ZG \tag{4.26}$$

where matrix Z is obtained by any pole placement method so that $F - ZG$ is stable. Matrices E and K are obtained according to

$$\begin{bmatrix} E & K \end{bmatrix} = L\bar{A}\Sigma^+ + Z(I - \Sigma\Sigma^+) \tag{4.27}$$

where $\bar{A} = A\,(I - L^+L)$, $\bar{C} = C\,(I - L^+L)$ and $\Sigma = \begin{bmatrix} C\bar{A} \\ \bar{C} \end{bmatrix}$, and matrix J is obtained according to

$$J = K + NE \tag{4.28}$$

whilst matrix H is obtained according to

$$H = (L - EC)\,B \tag{4.29}$$

By using this algorithm we can compute all the observer parameters which provide a functional observer of the form

$$\dot{w}(t) = Nw(t) + Jy(t) + Hu(t) \tag{4.30}$$

$$\hat{z}(t) = w(t) + Ey(t) \tag{4.31}$$

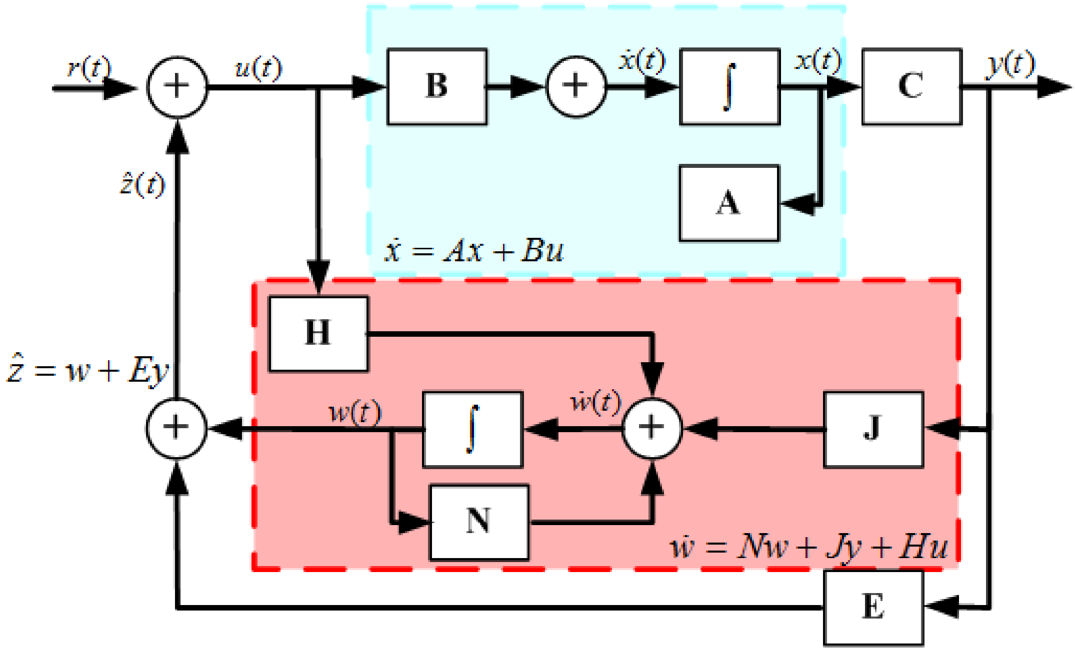

Figure 4.2: Exploded view of Darouach Functional Observer

The block diagram of the Darouach Functional Observer depicted above has three main areas, the blue area represents the system dynamics, the red area constructs some linear combination of the states $w(t)$ or equation (4.30), whilst the output of the summing function in the bottom left constructs the required state function or equation (4.31)

4.3.3 Summary

It is evident from the above-mentioned algorithm that no strict canonical forms are required. This makes the design attractive for implementation. The method allows for some control of the observer eigenvalues and guarantees stability of the observer system through condition (4.23). The eigenvalues of the observer are set according to the equation

$$N = F - ZG \tag{4.32}$$

where F and G are intermediate matrices defined in the algorithm. The matrix Z is arbitrary and is used to set the eigenvalues of the observer. Whilst the design method allows arbitrary eigenvalues, the exact choice for Z is not immediately obvious. Another attractive feature is that it can be readily extended to discrete time systems.

Chapter 5

Insulin Regulation Problem

Diabetes Mellitus is an incurable disease that is estimated to be affecting more than 180 million people worldwide. Characterised by the inability of the pancreas to regulate blood glucose concentration [36], diabetes can over time cause damage to the heart, blood vessels, eyes, kidneys, and nerves. Type 1 Diabetes Mellitus, also known as insulin-dependent diabetes, is characterised by the inability of the pancreas to release appropriate amounts of insulin. The hormone insulin metabolises glucose, which is then absorbed by the cells [37]. Conversely, Type 2 Diabetes Mellitus is a metabolic disorder characterised by insulin resistance, and as a result is also referred to as non insulin-dependent diabetes.

Inadequate secretion of insulin by the diabetic pancreas results in poor maintenance of normoglycemia, defined as the normal condition of blood glucose concentrations in the 70-100 mg/dL range [36]. After non-diabetic subjects have a meal, the glucose level increases, with a peak at 120-140 mg/dL for a period of two hours, followed by a decrease to normoglycemic level. In diabetic patients, the blood glucose level remains at a hyperglycemic level for a long period of time [37]. Therefore, external doses of insulin are systematically required to control the disease. However, over-delivery of exogenous insulin can cause hypoglycemia (defined as the condition of blood glucose concentrations less than 60 mg/dL) which can lead to insulin shock [36]. As a result, it is important to regulate a diabetic patient's blood glucose concentrations within tight physiological limits.

Currently, the treatment for insulin-dependent patients includes subcutaneous (under the skin) insulin treatment. This type of treatment may require up to four or five daily injections, usually corresponding with mealtimes. The amount of insulin injected is typically based on a glucose measurement (finger prick method), an approximation of the glucose content of the upcoming meal, and the estimated insulin release kinetics from the subcutaneous depot [36].

An alternative treatment solution comprises of a device that delivers insulin intravenously. The device consists of a mechanical pump, a glucose sensor that is either *in vivo* (inside a living organism) or measures glucose from the skin, and a mathematical algorithm which regulates the mechanical pump given the sensor measurement. Intravenous delivery of insulin has significant advantages over open-loop solutions such as injections or slow release preparations. This form of delivery features: (i) rapid delivery with negligible dead-time; (ii) a higher percentage of drug reaching the blood stream; (iii) faster response to insulin over-delivery; (iv) potential for improved closed-loop controller performance. In addition, instead of using a conventional finger prick method to monitor blood glucose, an implantable glucose sensor or skin sensor could be used, therefore increasing patient comfort and thus compliance with the treatment [36].

5.1 Type I Diabetic Patient Model

The human glucose-insulin model used in this dissertation is based on the initial works of Sorenson [38], which is then further modified and presented by Parker [36]. The model uses compartmental modelling techniques, where individual compartments are obtained by performing mass balances around tissues important to glucose or insulin dynamics. The principal advantage of a compartmental technique is that the model design is based on an understanding of physiology within the body. Consequently, numerical simulations can yield insight into the physiological parameters [39]. As depicted in Figure 5.1 there are six compartments isolating each organ, being the brain, heart/lungs, gut, liver, kidney, and periphery. The arrows indicate the flow of blood and each box represents a compartment. The model combines the stomach and intestinal effects into the gut compartment, and muscle and adipose tissue effects into the periphery compartment. Glucose is transported by the blood to various compartments via convection. Once in a compartment, glucose and insulin are either metabolised, or transported via diffusion into a tissue sub-compartment where they are subsequently metabolised.

The notation used for the model description is presented in Table 5.1.

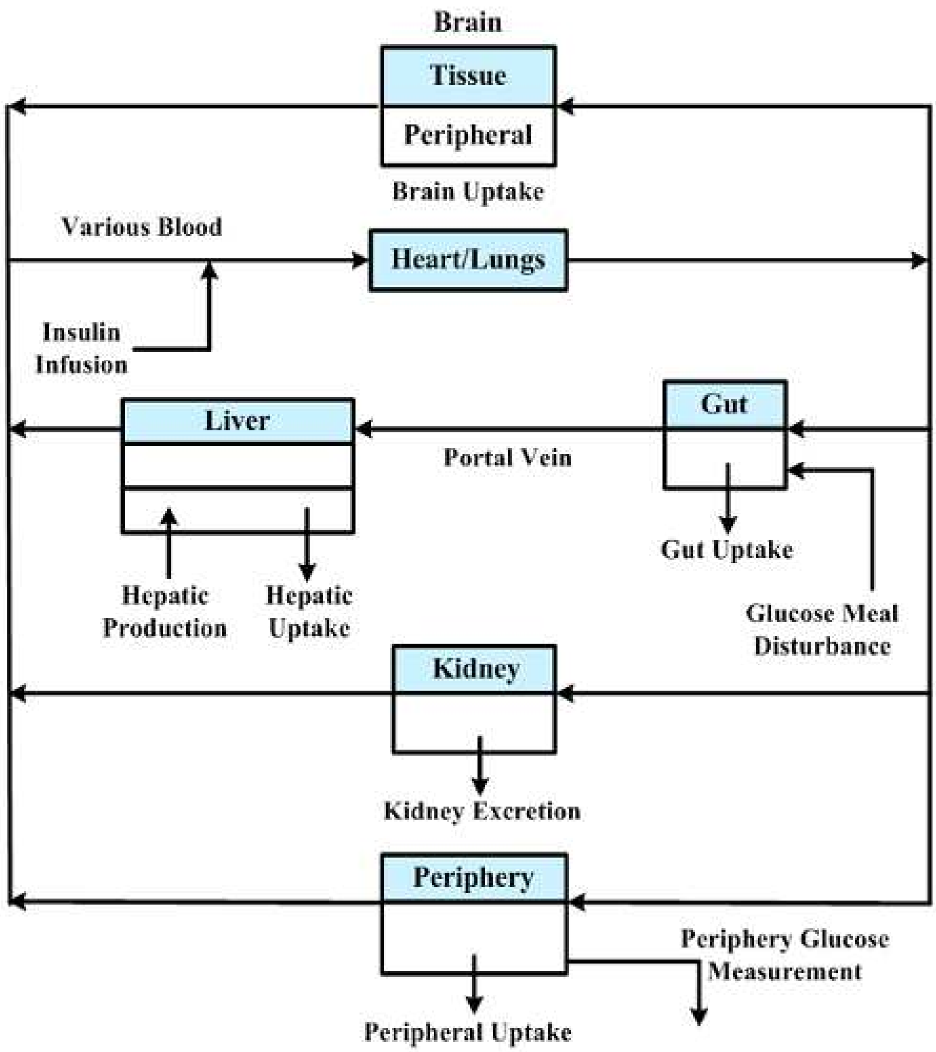

Figure 5.1: Compartmental Diagram of the Glucose and Insulin Systems in a Type 1 Diabetes Patient

Model Variables
A = auxiliary equation state (dimensionless)
B = fractional clearance (I, dimensionless; N, L/min)
G = glucose concentration (mg/dL)
I = insulin concentration (mU/L)
N = glucagon concentration (normalised, dimensionless)
Q = vascular plasma flow rate (L/min)
q = vascular blood flow rate (dL/min)
T = transcapillary diffusion time constant (min)
V = volume (L)
v = volume (dL)
Γ = metabolic source or sink rate(mg/min or mU/min)
Glucose Model Sub and Superscripts
A = hepatic artery
B = brain
BU = brain uptake
C = capillary space
G = glucose
H = heart and lungs
HGP = hepatic glucose production
HGU = hepatic glucose uptake
I = insulin
$IHGP$ = insulin effect on HGP
$IHGU$ = insulin effect on HGU
IVI = intravenous insulin infusion
K = kidney
KC = kidney clearance
KE = kidney excretion
L = liver
LC = liver clearance
N = glucagon
$NHGP$ = glucagon effect on HGP
P = periphery (muscle/adipose tissue)
PC = peripheral clearance
PGU = peripheral glucose uptake
PIR = pancreatic insulin release
PNC = pancreatic glucagon clearance
PNR = pancreatic glucagon release (normalised)
$RBCU$ = red blood cell uptake
S = gut (stomach/intestine)
SIA = insulin absorption into the blood stream from subcutaneous depot
SU = gut uptake
T = tissue space

Table 5.1: Notation for Type 1 Diabetes Model

The glucose submodel differential mass balance equations are given by

$$\dot{G}_B^C = (G_H^C - G_B^C)\frac{q_B}{v_B^C} - (G_B^C - G_B^T)\frac{v_B^T}{T_B v_B^C} \tag{5.1}$$

$$\dot{G}_B^T = (G_B^C - G_B^T)\frac{1}{T_B} - \frac{\Gamma_{BU}}{v_B^T} \tag{5.2}$$

$$\dot{G}_H^C = (G_B^C q_B + G_L^C q_L + G_K^C q_K + G_P^C q_P - G_H^C q_H - \Gamma_{RBCU})\frac{1}{v_H^C} \tag{5.3}$$

$$\dot{G}_S^C = (G_H^C - G_S^C)\frac{q_s}{v_S^C} + \frac{\Gamma_{meal}}{v_S^C} - \frac{\Gamma_{su}}{v_S^C} \tag{5.4}$$

$$\dot{G}_L^C = (G_H^C q_A + G_S^C q_S - G_L^C q_L)\frac{1}{v_L^C} + \frac{\Gamma_{HCP}}{v_L^C} - \frac{\Gamma_{HGU}}{v_L^C} \tag{5.5}$$

$$\dot{G}_K^C = (G_H^C - G_K^C)\frac{q_K}{v_K^C} - \frac{\Gamma_{KE}}{v_K^C} \tag{5.6}$$

$$\dot{G}_P^C = (G_H^C - G_P^C)\frac{q_P}{v_P^C} + (G_P^T - G_P^C)\frac{v_P^T}{T_P^G v_P^C} \tag{5.7}$$

$$\dot{G}_P^T = (G_P^C - G_P^T)\frac{1}{T_P^G} - \frac{\Gamma_{PGU}}{v_P^T} \tag{5.8}$$

The metabolic source and sink terms (Γ_i [=] mg/min) in the above equations are defined by

$$\Gamma_{BU} = 70 \tag{5.9}$$

$$\Gamma_{RBCU} = 10 \tag{5.10}$$

$$\Gamma_{SU} = 20 \tag{5.11}$$

$$\Gamma_{HGP} = 155 A_{IHGP}(2.7\tanh(0.388N) - A_{NHGP}) \times [1.425 - 1.406\tanh 0.6199(\frac{G_L^C}{101} - 0.4969)] \tag{5.12}$$

$$\dot{A}_{IHGP} = \frac{1}{25}[1.2088 - 1.138\tanh(1.669\frac{I_L^C}{21.43} - 0.8885) - A_{IHGP}] \tag{5.13}$$

$$\dot{A}_{NHGP} = \frac{1}{65}[\frac{2.7\tanh(0.388N) - 1}{2} - A_{NHGP}] \tag{5.14}$$

$$\Gamma_{HGU} = 20 A_{IHGU}[5.6648 + 5.6589\tanh(2.4375(\frac{G_L^C}{101} - 1.48))] \tag{5.15}$$

$$\dot{A}_{IHGU} = \frac{1}{25\text{min}}[2\tanh(0.549\frac{I_L^C}{21.43}) - A_{IHGU}] \tag{5.16}$$

$$\Gamma_{KE} = \begin{cases} 71 + 71\tanh[0.011(G_K^C - 460)] & \text{for } G_K^C < 460 \text{ mg/dL} \\ 0.872 G_K^C - 300 & \text{for } G_K^C \geq 460 \text{ mg/dL} \end{cases} \tag{5.17}$$

$$\Gamma_{PGU} = \frac{35 G_P^T}{86.81}[7.035 + 6.51623\tanh 0.33827(\frac{I_P^T}{5.304} - 5.82113)] \tag{5.18}$$

(5.19)

The insulin sub-model mass balances are given by

$$\dot{I}_B^C = (I_H^C - I_B^C)\frac{Q_B}{V_B^C} \quad (5.20)$$

$$\dot{I}_H^C = (I_B^C Q_B + I_L^C Q_L + I_K^C Q_K + I_P^C Q_P - I_H^C Q_H - \Gamma_{IVI})\frac{1}{V_H^C} \quad (5.21)$$

$$\dot{I}_S^C = (I_H^C - I_S^C)\frac{Q_S}{V_S^C} \quad (5.22)$$

$$\dot{I}_L^C = (I_H^C Q_A + I_S^C Q_S - I_L^C Q_L)\frac{1}{V_L^C} + \frac{\Gamma_{PIR} - \Gamma_{LC}}{V_L^C} \quad (5.23)$$

$$\dot{I}_K^C = (I_H^C - I_K^C)\frac{Q_K}{V_K^C} - \frac{\Gamma_{KC}}{V_K^C} \quad (5.24)$$

$$\dot{I}_P^C = (I_H^C - I_P^C)\frac{Q_P}{V_P^C} - (I_P^C - I_P^T)\frac{V_P^T}{T_P^I V_P^C} \quad (5.25)$$

$$\dot{I}_P^T = (I_P^C - I_P^T)\frac{1}{T_P^I} + \frac{\Gamma_{SIA} - \Gamma_{PC}}{V_P^T} \quad (5.26)$$

$$(5.27)$$

The related metabolic sinks (Γ_i[=] mU/min) are

$$\Gamma_{LC} = F_{LC}(I_H^C Q_A + I_S^C Q_S + \Gamma_{PIR}) \quad (5.28)$$

$$\Gamma_{PIR} = 0, \text{ no pancreatic insulin release} \quad (5.29)$$

$$\Gamma_{KC} = F_{KC} I_K^C Q_K \quad (5.30)$$

$$\Gamma_{PC} = \frac{I_P^T}{1 - F_{PC}} \quad (5.31)$$

Glucagon, the potentiator responsible for stimulating glucose release into the bloodstream, is modelled as a single blood pool compartment, governed by the mass balance

$$\dot{N} = (\Gamma_{PNR} - N)\frac{F_{PNC}}{V_N} \quad (5.32)$$

Glucagon release from the α-cells of the pancreas (Γ_i[=]μg/min) is defined by

$$\Gamma_{PNR} = [1.3102 - 0.61016 \tanh 1.0571(\frac{I_H^C}{15.15} - 0.46981)] \times [2.9285 - 2.095 \tanh 4.18(\frac{G_H^C}{91.89} - 0.6191)] \quad (5.33)$$

Overall, the diabetic patient is modelled by nineteen differential equations, eleven of those describing glucose dynamics, seven describing insulin dynamics, and a single one describing the compartment for glucagon.

5.2 Linearisation of Diabetic Insulin-Glucose Model

From the diabetic patient model presented in Section 5.1, it is clear that this is a non-linear model. The question of how to deal with the non-linear model in regulating blood-glucose levels in the diabetic patient thus arises. Motivated by the fact that functional observers are based on linear systems, one possibility is to make a linearised approximation to the non-linear model, allowing us to apply linear theory. This requires us to provide a nominal operating point from which we can linearise the model.

5.2.1 Operating Point

The system output is the peripheral tissue glucose concentration (i.e. the glucose concentration of interstitial fluid that surrounds an individual cell in the body), and it is represented by equation (5.8). The use of these glucose measurements provides accurate blood glucose levels. The control input is the insulin entering the patient through an intravenous route, which is modelled by equation (5.21). The model also incorporates a second input, corresponding to a meal, which for control formulation is seen like a disturbance input in equation (5.4). The meal can be simulated by the model presented in Lehmann and Deutsch [40].

As mentioned in Section 2.1, a nominal operating point from which the Jacobian matrices can be evaluated is required in order to form our system matrices. The nominal operating point is chosen to represent a healthy human, the idea being that the controller will regulate the patient about this operating point. The nominal operating point is based on an insulin delivery rate of 22.3 mU/min [41], and on meal disturbances having a nominal value of 0 mg/min (absorption into the bloodstream).

Using these nominal inputs, we take the non-linear model and model the steady-state values of the state variables to construct a state vector at equilibrium. From these values we can then evaluate the appropriate Jacobian to form our 19^{th} order linearised system model.

Evaluating the appropriate Jacobian matrix at the equilibrium states will yield a state-space system of the form

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{5.34}$$

$$y(t) = Cx(t) \tag{5.35}$$

The steady-state values provided by the non-linear models are as follows

Equilibrium steady state values x_0 for 19^{th} order non-linear model			
1) $G_B^C = 75.7804$	2) $G_B^T = 43.1137$	3) $G_H^C = 87.6448$	4) $G_S^C = 85.6646$
5) $G_L^C = 97.9247$	6) $G_K^C = 87.6409$	7) $G_P^C = 84.3674$	8) $G_P^T = 80.4398$
9) $A_{IHGP} = 1.1593$	10) $A_{NHGP} = -0.0997$	11) $A_{IHGU} = 0.9276$	12) $I_B^C = 32.6663$
13) $I_H^C = 32.6663$	14) $I_S^C = 32.6663$	15) $I_L^C = 19.5998$	16) $I_K^C = 22.8664$
17) $I_P^C = 84.3674$	18) $I_P^T = 11.4332$	19) $N = 0.7880$	

Table 5.2: Steady State Values from Non-Linear Model

The system matrices A, B, and C are presented as follows

Elements of System Matrix $B \in \mathbb{R}^{19\times 1}$ matrix $B_{i,j}$ refers to the i^{th} row and j^{th} column element in matrix B	
$B_{13,1} = \frac{-1}{V_H^C} = -1.0152$	(All other unspecified matrix elements are zero)

Table 5.3: Linearised B Matrix

Elements of System Matrix $C \in \mathbb{R}^{1\times 19}$ matrix $C_{i,j}$ refers to the i^{th} row and j^{th} column element in matrix C	
$C_{1,8} = 1$	(All other unspecified matrix elements are zero)

Table 5.4: Linearised C Matrix

Elements of System Matrix $A \in \mathbb{R}^{19\times19}$ **matrix**
$A_{i,j}$ refers to the i^{th} row and j^{th} column element in matrix A

$A_{1,1} = -2.298$	$A_{1,2} = 0.6122$	$A_{1,3} = 1.6857$	
$A_{2,1} = 0.4762$	$A_{2,2} = -0.4762$		
$A_{3,1} = 0.4275$	$A_{3,3} = -3.1667$	$A_{3,5} = 0.913$	$A_{3,6} = 0.7319$
$A_{3,7} = 1.0942$			
$A_{4,3} = 0.9018$	$A_{4,4} = -0.9018$		
$A_{5,3} = 0.0996$	$A_{5,4} = 0.4024$	$A_{5,5} = -0.5817$	$A_{5,9} = 5.6958$
$A_{5,10} = -7.3338$	$A_{5,11} = -0.6961$	$A_{5,19} = 7.0072$	
$A_{6,3} = 1.5303$	$A_{6,6} = -1.5304$		
$A_{7,3} = 1.4519$	$A_{7,7} = -2.7481$	$A_{7,8} = 1.2962$	
$A_{8,7} = 0.200$	$A_{8,8} = -0.2091$	$A_{8,18} = -0.057$	
$A_{9,9} = -0.04$	$A_{9,15} = -0.0024$		
$A_{10,10} = -0.0154$	$A_{10,19} = 0.0073$		
$A_{11,11} = -0.04$	$A_{11,15} = 0.0016$		
$A_{12,12} = -1.6981$	$A_{12,13} = 1.6981$		
$A_{13,12} = 0.4569$	$A_{13,13} = -3.1675$	$A_{13,15} = 0.9137$	$A_{13,16} = 0.731$
$A_{13,17} = 1.066$			
$A_{14,13} = 0.7619$	$A_{14,14} = -0.7619$		
$A_{15,13} = 0.0947$	$A_{15,14} = 0.3789$	$A_{15,15} = -0.7895$	
$A_{16,13} = 1.4257$	$A_{16,16} = -1.8535$		
$A_{17,13} = 1.4286$	$A_{17,17} = -1.8571$	$A_{17,18} = 0.4286$	
$A_{18,17} = 0.05$	$A_{18,18} = -0.0795$		
$A_{19,3} = -0.0014$	$A_{19,13} = -0.0004$	$A_{19,19} = -0.0916$	

(All other unspecified matrix elements are zero)

Table 5.5: Linearised A Matrix

5.3 Model Reduction

From the linearised system matrices, we find that the system is neither controllable nor observable. From a controllability point of view, this implies that certain compartments in the insulin-glucose model are physically isolated from intravenous infusion. This is reasonable, as it is unlikely that insulin alone will be able to drive all states to desired values, as certain states are a function of other physiological processes decoupled from the input. From an observability point of view, this implies that information about all states of the system cannot be deduced from the sensed glucose measurement alone. The system must be minimal in order for a controller and functional observer to be designed.

To overcome this problem we apply similarity transformations to separate out controllable and observable states. It then becomes important to ask what can be done about the non-controllable states. If these modes are stable then it may not be important that their eigenvalues are unchanged. In fact, we can verify that the eigenvalues of the uncontrollable partition are stable, and as a result that the system as a whole can be stabilised.

Knowing the number of controllable and observable states, we then apply a balanced model reduction to provide a system that is both minimal and that has similar input-output characteristics as the original system. The similarity transforms provide us with nine states that are both controllable and observable. Accordingly, this is the order that our model should be reduced to. The resulting reduced order system must preserve the required input-output properties and must also be a good approximation of the full order equivalent for appropriate control design.

Using Matlab we apply *balanced model reduction* to obtain a 9^{th} order model. Although this process produces a minimal system, due to the transformations involved in the model reduction process (see Section 2.4), it results in the loss of biological meaning. The original model has a good biological map, where each individual state has its own biological significance. The reduced-order model on the other hand, is strictly mathematical, the states lose their biological meaning thereby making the model 'non-biological'. Most importantly, from a control design point of view, the reduced-order model retains its input-output biological correspondence, in that the dynamics of the lower-order states, once scaled to an output (through $y(t) = Cx(t)$), return meaningful values for the biological quantities of interest. Therefore, provided we are not overly concerned with meaning of the states, a functional observer can be used to supply the control law to stabilise the system.

Further details of the reduced-order system matrices can be obtained by running the m.file in Matlab (*glucosemodelreduction.m*). From this point onwards, all analysis will be based on the model reduced system matrices. The observability and controllability matrices are now of full rank, allowing us to proceed with controller and observer design. Possible future work could test for controllability and observability of different arbitrary operating points to gain further knowledge of the non-linear model.

Figure 5.2 presents the frequency response of the full-order and reduced-order SISO models. The region of interest is the bandwidth region, defined as the portion of frequency spectrum in which control is effective. The process model begins to exhibit behavioural changes for frequencies greater than 100 rad/sec. This is sufficient to capture all the full-order behaviour as we only expect the fastest disturbances affecting the patient to have a time constant of approximately five minutes [41]. In fact, it may be possible to reduce the model further whilst still retaining sufficient accuracy, and should be a topic of future work.

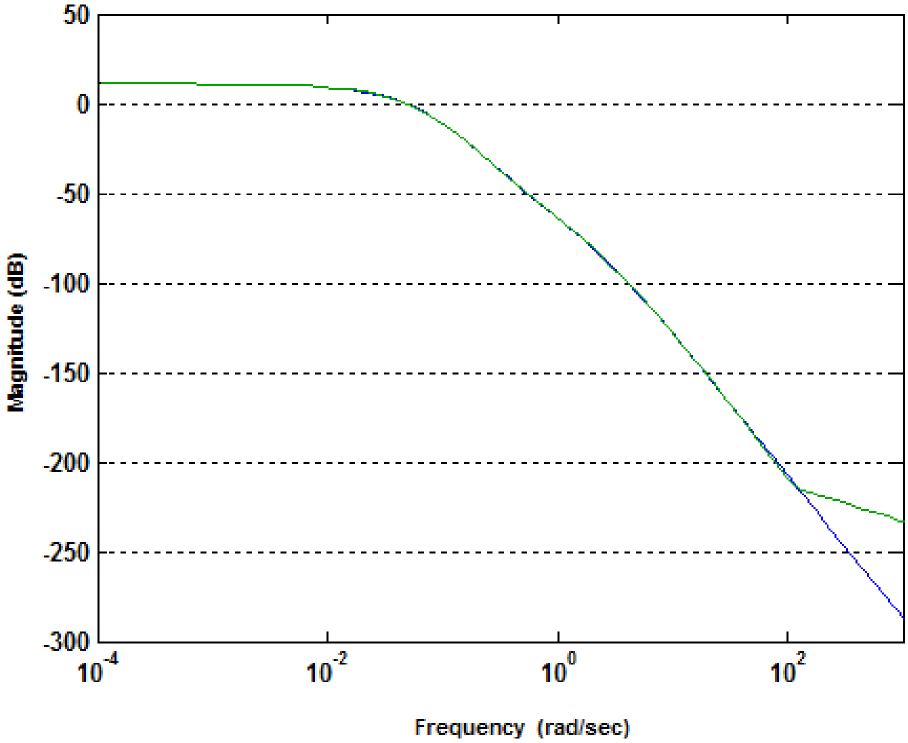

Figure 5.2: Effect of Model Reduction as a Function of Frequency

5.4 Controller Design

Having linearised the system to produce a state space model of the form in equation (5.34), the linear model can be used to analyse the system characteristics of the glucose-insulin model. One important characteristic, that will provide background information about the system as well as information for designing the controller, is the open-loop system response. The open-loop linearised system's eigenvalues of system matrix A are

$$\text{eig}(A) = \begin{bmatrix} -4.6168 \\ -0.8197 + 0.2692i \\ -0.8197 - 0.2692i \\ -0.2385 \\ -0.1745 \\ -0.0098 \\ -0.0769 \\ -0.0401 \\ -0.0489 \end{bmatrix} \tag{5.36}$$

Since all the eigenvalues of the system have negative real parts, the system is stable. The open-loop response of the system is presented in Figure 5.3.

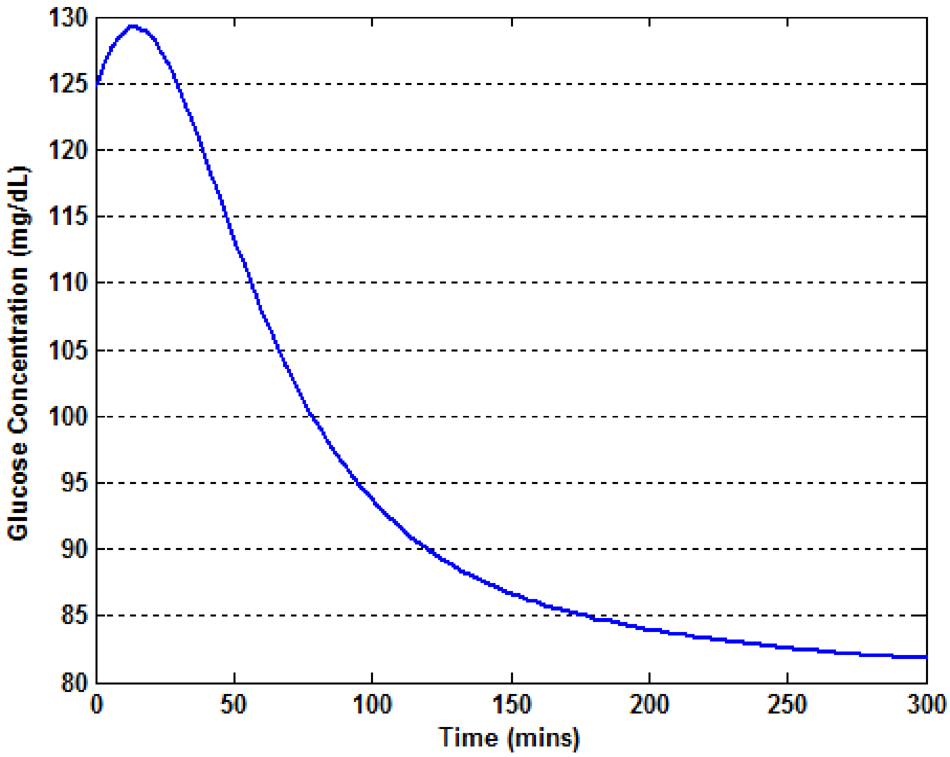

Figure 5.3: Open-loop Response of Glucose-Insulin Model

The sensed output, on the vertical axis, presents a patient's glucose level outside the normoglycemic range (ie. after a meal). Recall that the nominal operating point is based on an insulin infusion rate of 22.3 mU/min. A diabetic patient would have glucose levels that remain high for extended periods of time after a meal, so the closed-loop system should aim to drive the glucose to normoglycemic levels in a comparatively short period of time.

5.4.1 Pole Placement Design

A controller based on pole placement can be used to improve the system characteristics such that closed-loop performance will satisfy the required criteria: namely that the glucose level be returned to normal levels in comparatively less time.

Recall from Section 2.5.1 that in choosing new pole locations, the general philosophy is to minimise control effort (in this case insulin input), whilst satisfying required design criteria, in this case providing normoglycemic conditions for the patient. From the eigenvalues of the open-loop system presented in (5.36), the sixth element is relatively unstable. We shift this pole 0.02 to the left to yield the following control law

$$L = \begin{bmatrix} 0.0976 & 0.0012 & 0.0963 & -0.0397 & -0.0016 & -0.0033 & 0.0016 & 0.0002 & -0.0001 \end{bmatrix}$$

This yields the following closed-loop response and insulin control effort

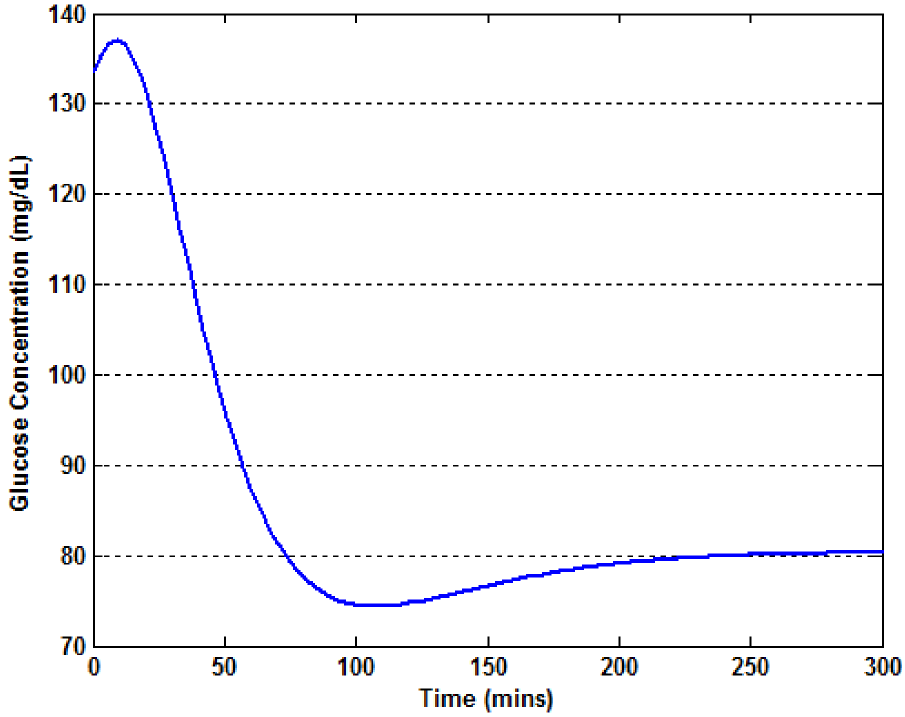

Figure 5.4: Close-loop Response of Pole Assignment Scheme 1

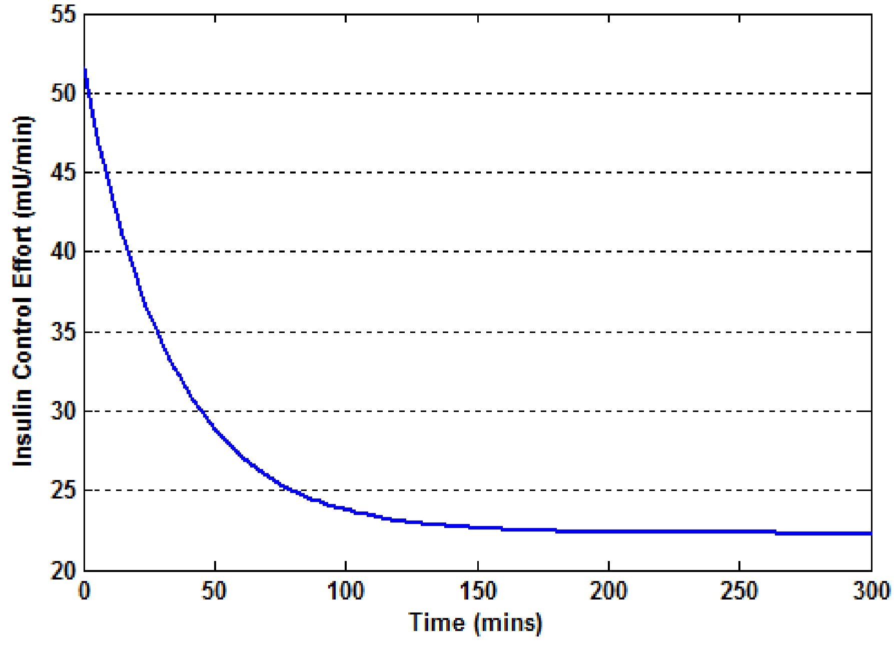

Figure 5.5: Insulin Control Effort for Pole Assignment 1

The graph of the closed-loop response of Figure 5.4 shows significant overshoot past the nominal operating point of 80.4 mg/dL. Whether this overshoot would adversely affect the patients' short or long term health is beyond the scope of this dissertation. It would be reasonable to expect that a healthy persons' glucose level would fluctuate within normoglycemic range, nonetheless information on this issue should be readily available. If an overshoot cannot be allowed, an alternate set of pole locations must be found.

By leaving the sixth element and moving elements seven to nine to the left by 0.05 we obtain a feedback gain matrix of

$$L = \begin{bmatrix} 0.5042 & -0.3576 & -0.4314 & 0.2069 & 0.0219 & -0.0116 & 0.0110 & 0.0023 & -0.0012 \end{bmatrix}$$

This yields the closed-loop response in Figure 5.6

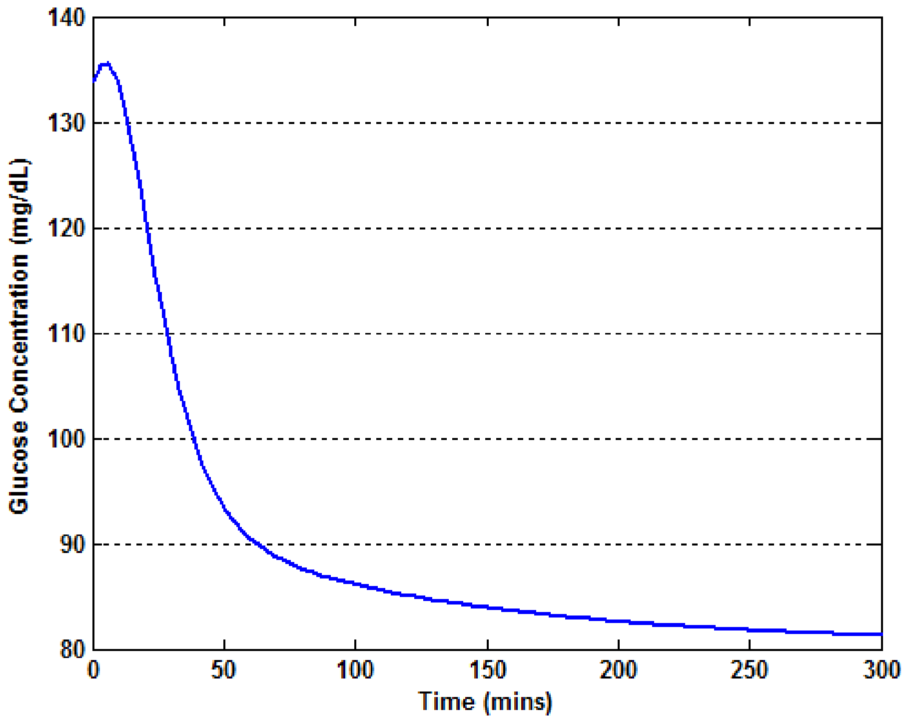

Figure 5.6: Close-loop Response of Pole Assignment Scheme 2

The new pole choices have eliminated the overshoot and the system is critically damped. In terms of control effort the first solution is more efficient in the usage of insulin but suffers from overshoot. This pole configuration will be used to analyse the control effort required to drive the glucose concentration back to desired values.

It is also important to note that the theory allows for the control law to produce a negative input. Clearly in this particular application, insulin cannot be drawn out of the patient.

This problem could be overcome by setting any negative values to zero. Possible future work could model the effect of this.

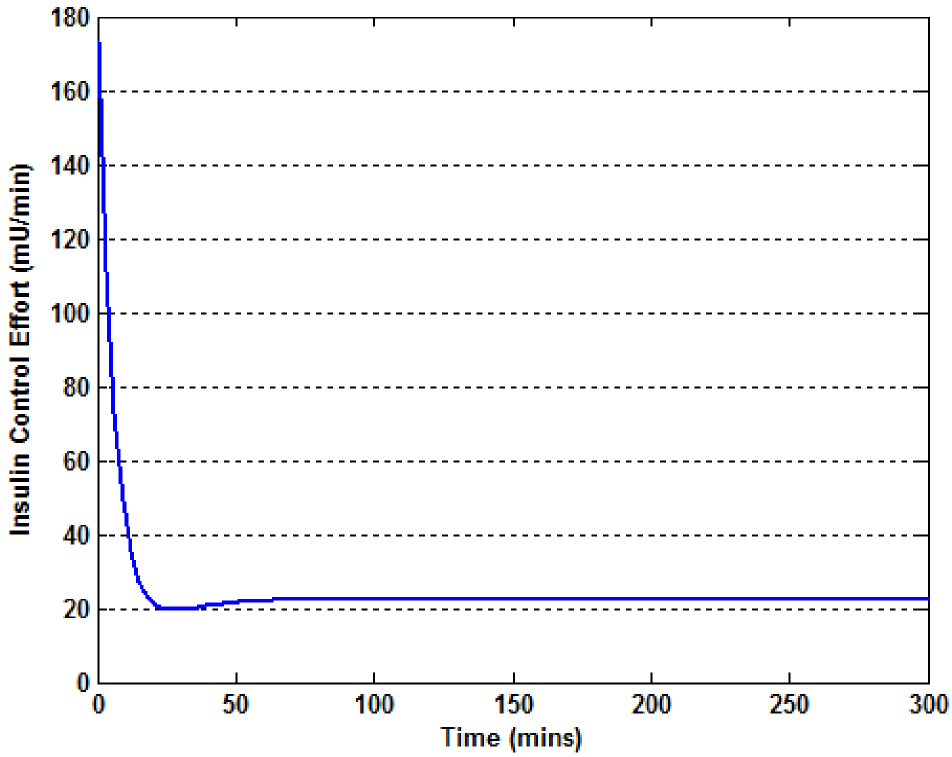

Figure 5.7: Insulin Control Effort for Pole Assignment 2

Figure 5.7 describes the control effort required to drive the glucose dynamics back to nominal values for the second pole placement scheme. There is a slight overshoot, the control effort reverts back to its nominal 22.3 mU/min operating point in steady-state. Comparing Figure 5.6 and 5.7, the dynamics of the output and input are very similar with some time-delay.

5.4.2 Linear Quadratic Regulator

The LQR design involves selecting matrices Q and R such that the performance of the closed-loop system regulates glucose levels within the physiological range mentioned earlier. From Section 2.5.2, matrix Q determines the relative importance of the state values deviating from zero, and matrix R determines the relative importance of the input u (insulin infusion) deviating from zero.

A problem associated with designing the Q matrix is that it requires knowledge of individual state information. This makes it difficult to design the R and Q matrices (see Section 2.5.2) for the regulator, as it is not clear what is being optimised. It is nonetheless possible to obtain a transformation matrix which identifies the makeup of the new states. These new states are a linear combination of the old states and, if appropriate, can be optimised by the regulator.

Another way around the problem is to consider optimising the output exclusively. This approach only requires us to appreciate the biological meaning of the output (which we do). Consider the cost function J from equation (2.34)

$$J = \int_0^\infty [x^T(t)Qx(t) + u^T(t)Ru(t)]dt \tag{5.37}$$

The first term of the integrand optimises state energy, a Q matrix needs to be designed that optimises the output. Now consider the output equation

$$y(t) = Cx(t) \tag{5.38}$$

Modifying cost function J from (5.37) and substituting (5.38) gives

$$\begin{aligned} J &= \int_0^\infty [y^T(t)y(t) + u^T(t)Ru(t)]dt \\ &= \int_0^\infty [x^T(t)C^TCx(t) + u^T(t)Ru(t)]dt \end{aligned}$$

From the above result, this solves our matrix Q such that

$$Q = C^TC$$

The eigenvalues for matrix Q are then

$$\text{eig}(Q) = \begin{bmatrix} 0.0000 \\ 0.0000 \\ 0.0000 \\ 0.0000 \\ 0.0000 \\ 0.0000 \\ 0.0000 \\ 0.0000 \\ 0.0802 \end{bmatrix} \tag{5.39}$$

which verifies that Q is positive-semidefinite.

The matrix R on the other hand can be readily solved since there is only one input for which we are optimising. Matrix R is a 1×1 matrix, the value of which will determine the relative importance of input energy (insulin) optimisation. The next issue is iterating for values of R to obtain satisfactory closed-loop performance. We trial for R being values of one and five.

For a matrix selection of $Q = C^T C$ and $R = [1]$, we obtain a feedback gain matrix of

$$L = \begin{bmatrix} 0.1578 & -0.0140 & -0.0154 & -0.0014 & -0.0003 & -0.0002 & 0.0001 & 0.0000 & 0.0000 \end{bmatrix}$$

which yields the closed-loop response of Figure 5.8

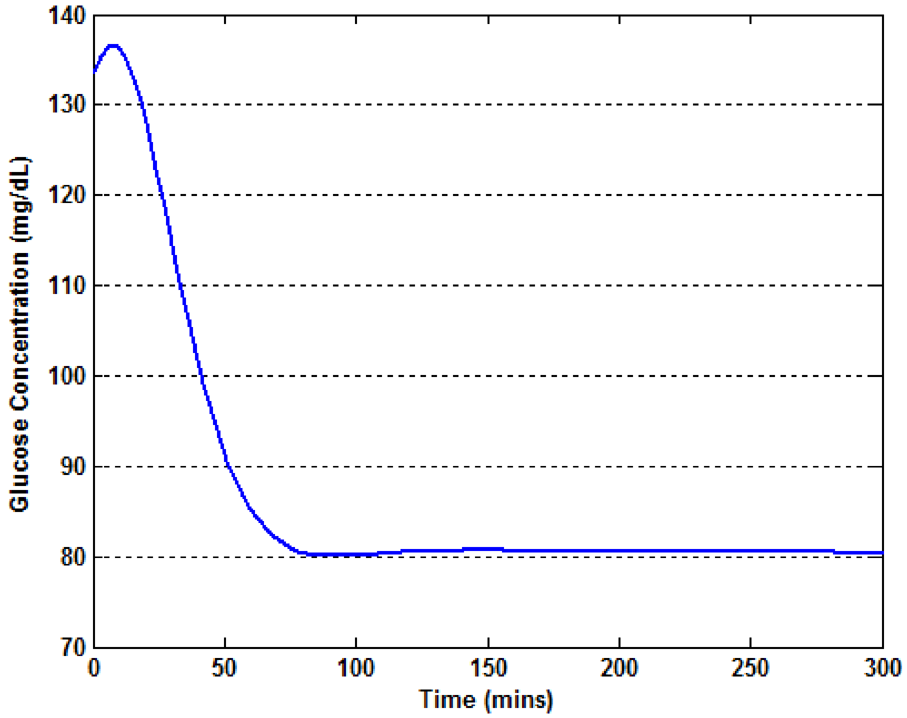

Figure 5.8: Close-loop Response of LQR with $R = 1$

The LQR controller returns the glucose level back to nominal values in just over an hour with practically no overshoot, this is a marked improvement in the rise time relative to that achieved by the pole placement (scheme 2) method.

The insulin effort against time is described by Figure 5.9

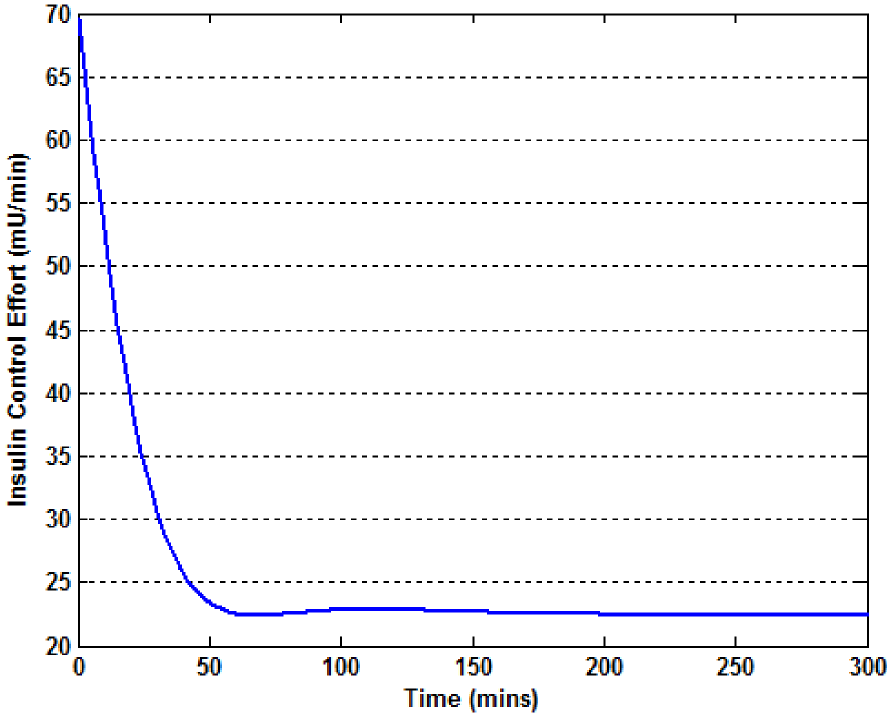

Figure 5.9: Insulin Control Effort for LQR with $R = 1$

For a matrix selection of $Q = C^T C$ and $R = [5]$, we obtain a feedback gain matrix of

$$L = \left[\begin{array}{ccccccccc} 0.0512 & -0.0090 & -0.0025 & 0.0003 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \end{array} \right]$$

and yields the closed-loop response of Figure 5.10.

The insulin effort against time is described by Figure 5.11.

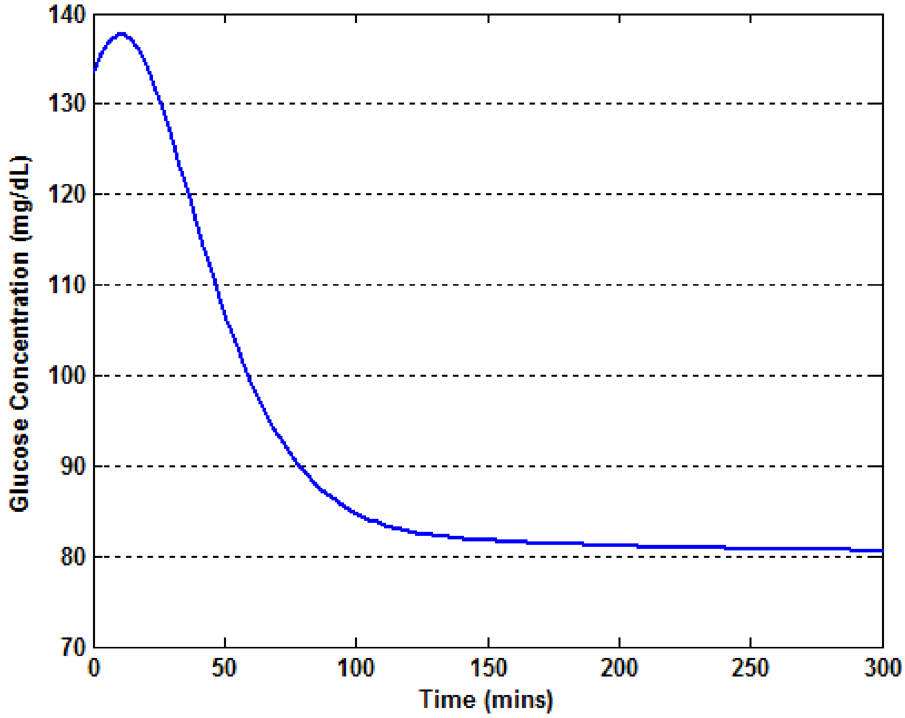

Figure 5.10: Close-loop Response of LQR with $R = 5$

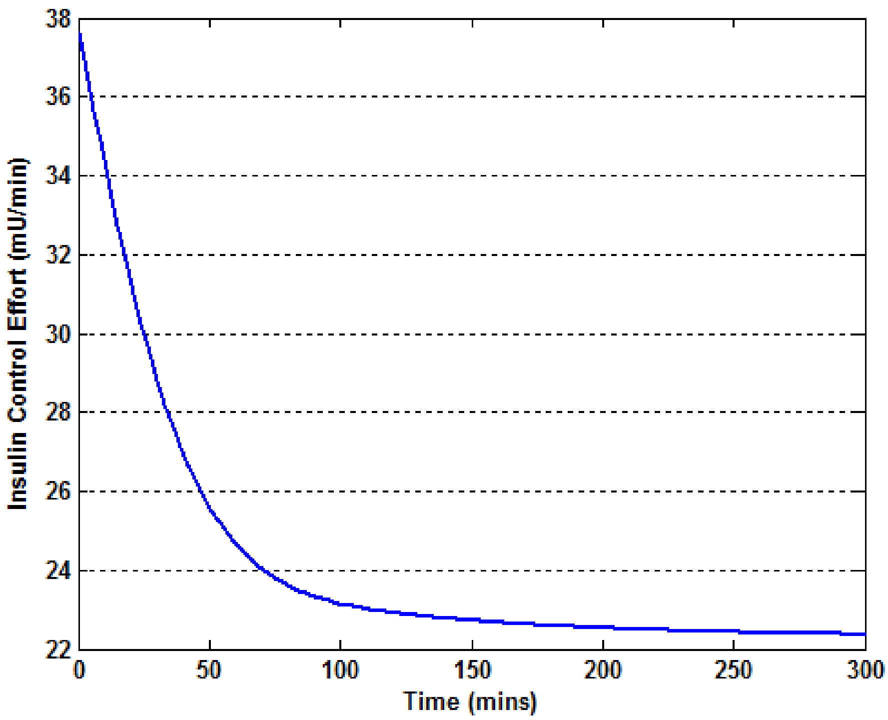

Figure 5.11: Insulin Control Effort for LQR with $R = 5$

5.4.3 Comparison of Different Design Methods

From Section 2.5, it was explained that both pole placement and the LQR approach are valid techniques in designing the controller for the glucose regulator. In order to determine which is best suited to the application we need to conduct some performance comparisons between the two methods.

To help determine the suitability of both methods, we should recall the objective of the glucose regulator. The main purpose of the regulator is to return the patients' blood glucose levels within normoglycemic levels comparatively faster than without a controller whilst also minimising two input parameters: (i) total insulin used, and (ii) the maximum insulin infusion rate. Maintaining glucose levels within normoglycemia ranges is important for the long term health of the patient. Since insulin is a consumable and requires replenishing, another objective of the design is to minimise the total insulin used. Further, by minimising the insulin infusion rate, smaller components can be used (ie. mechanical pump), thus decreasing the size of the implantable unit.

From inspecting Figures 5.12 and 5.13 we can notice some trends. There is a trade-off between how fast the glucose concentrations are returned to normoglycemic levels and the amount of insulin used to achieve the closed-loop response. LQR ($R = 5$) uses the least insulin and has the softest transient response. Pole placement (scheme 2) has the sharpest initial transient response but also requires the highest rate of insulin infusion, and therefore may require a larger mechanical pump. Although its initial transient response is sharpest, after approximately an hour, scheme 1 slows significantly while the other pole choices continue to drive glucose levels down. It seems that pole placement (scheme 1) and LQR ($R = 1$) are the most suitable for the application, although scheme 1 does suffer from significant overshoot. Therefore, the LQR method with matrix R set to 1 offers the best trade-off between shortest settling time, minimal use of insulin and negligible overshoot.

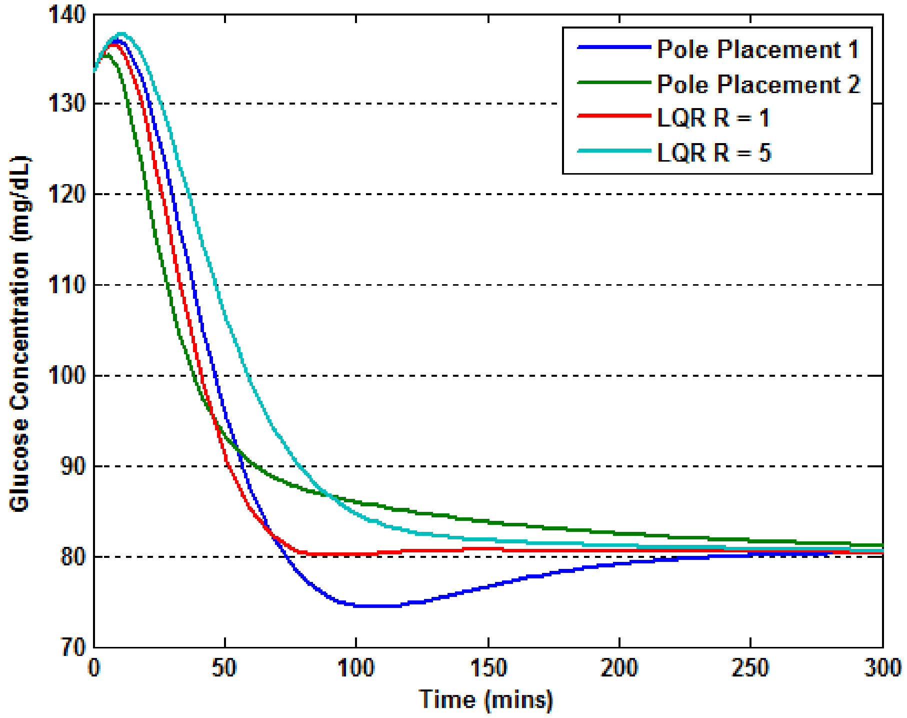

Figure 5.12: Comparison of Closed-loop Response

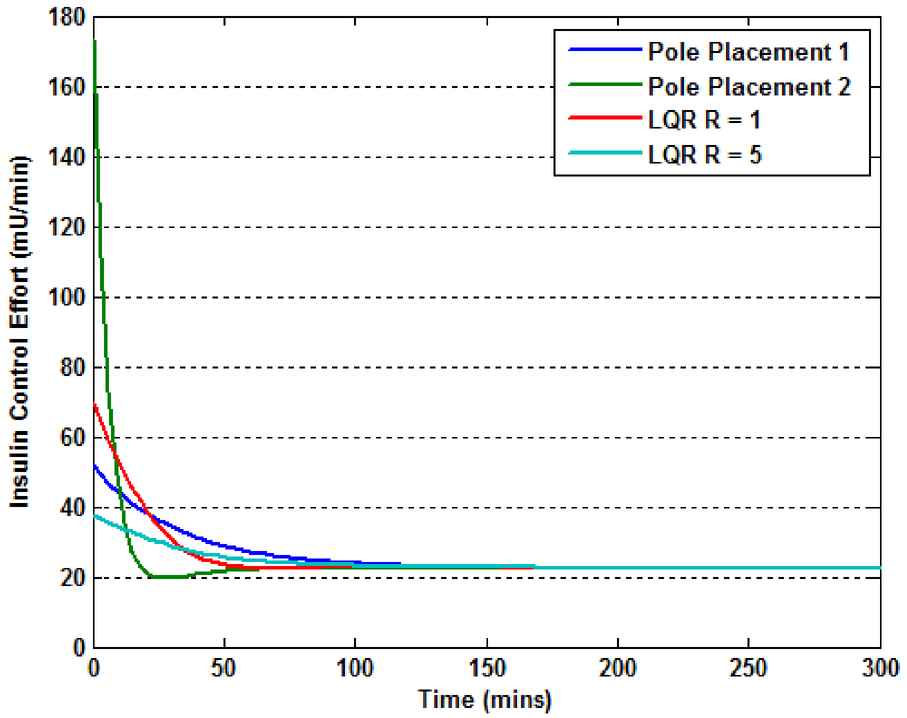

Figure 5.13: Comparison of Insulin Control Effort

5.5 Darouach Functional Observer

Having designed the linear control law, let us consider how to provide the state vector for feedback. A starting point is to measure those states. However as the state space was transformed by model reduction, the usefulness of sensors is limited. Further, even if the appropriate sensors were available, a patient may object to having numerous sensors being attached ex vivo and/or in vivo. This increases the chance of treatment being rejected.

The discussion of observer theory in Chapters 3 and 4 provides designers with numerous alternatives for estimating the state vector. To demonstrate the order-reducing capabilities of the functional observer, the Darouach functional observer will be used to directly estimate the linear control law.

5.5.1 Existence Conditions - A New Result

From Section 4.3.2, the necessary and sufficient condition for the existence of the Darouach functional observer is given by the following lemma

$$\text{rank}\begin{bmatrix} LA \\ CA \\ C \\ L \end{bmatrix} = \text{rank}\begin{bmatrix} CA \\ C \\ L \end{bmatrix} \tag{5.40}$$

The system and gain matrices violate this Lemma with the LHS(4) $\neq$ RHS(3). Clearly, the existence conditions for a minimal order observer cannot be met, and therefore the design of the observer cannot proceed.

There is however, a result that can overcome this problem; although at the expense of obtaining a minimal order observer. This involves designing a higher-order observer which may still be of lower order relative to a reduced-order observer. Designing a higher-order observer implies estimating additional states. If we append some matrix R to the feedback gain vector L then the existence conditions will become

$$\text{rank}\begin{bmatrix} LA \\ RA \\ CA \\ C \\ L \\ R \end{bmatrix} = \text{rank}\begin{bmatrix} CA \\ C \\ L \\ R \end{bmatrix} \tag{5.41}$$

where R can be chosen such that the RHS is of full rank n, which also implies that the LHS is of full rank n, thus satisfying the new condition.

To ensure that R is linearly independent of L, the form of R is defined as

$$R = \text{null}\begin{bmatrix} CA \\ C \\ L \end{bmatrix} \tag{5.42}$$

This is appended to L forming $\bar{L}$ as follows

$$\bar{L} = \begin{bmatrix} L \\ R \end{bmatrix} \tag{5.43}$$

which results in

$$\text{rank}(R) + \text{rank}\begin{bmatrix} CA \\ C \\ L \end{bmatrix} = n \tag{5.44}$$

This makes both sides of Lemma 5.40 equal to full rank n and therefore satisfies the necessary and sufficient condition for the existence of the observer. In this case, n is equal to nine, and it yields an observer of order seven. This is still smaller than the order achievable through a reduced-order observer (see Section 3.1.2).

The performance of the functional observer can be tested by plotting the error dynamics of the error function in equation (4.10). This error function presented in Figure 5.14 should asymptotically approach zero in finite time.

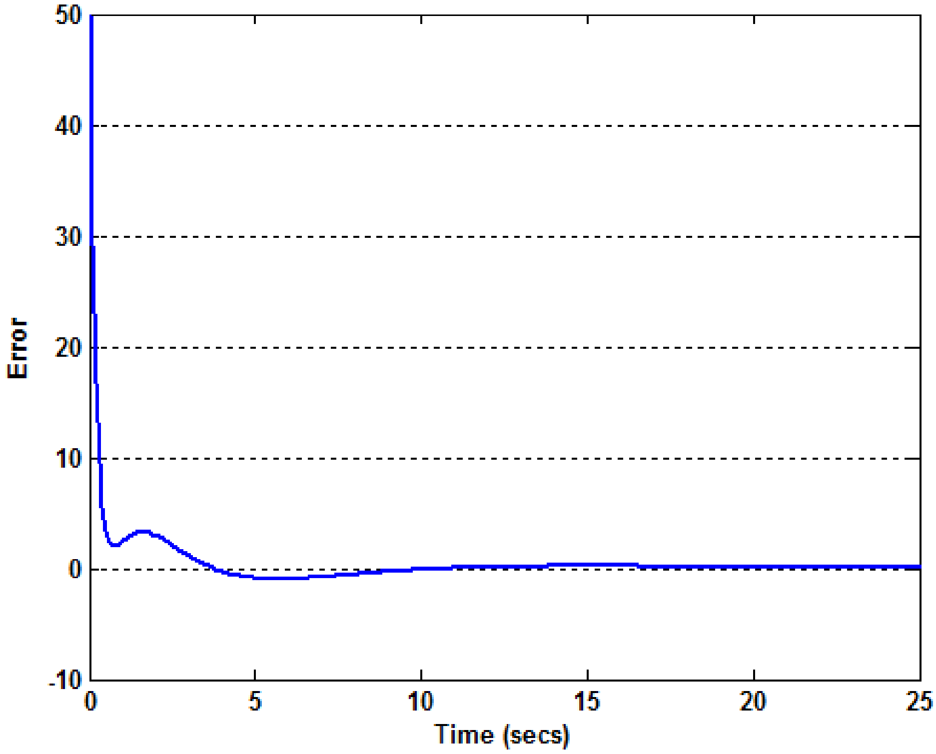

Figure 5.14: Observer Error of Functional Observer

The zero asymptotic error ensures that the control law estimate tends towards the correct function of states and ensures intended closed-loop dynamics. This completes the design process and successfully demonstrates the functional observer being used in a real world application. Its complexity is reduced relative to other known observers, namely the reduced-order observer, and its performance is demonstrated to be satisfactory. This functional observer can therefore be used in place of a reduced-order observer.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

Although the primary objective of this dissertation was to apply a Darouach functional observer in a real world problem, the application required us to consider a number of issues. The conclusions drawn may be of assistance to students who undertake similar projects in the future.

6.1.1 Linearisation

The system model that we used to describe glucose-insulin dynamics is non-linear, whilst observer theory is based on linear time-invariant systems. It was therefore necessary to linearise the system. This required a nominal operating point, which was achieved by simulating the non-linear model for some nominal input and then taking the steady-state values of the individual states to form a state vector. The Jacobian matrices can then be evaluated.

6.1.2 Model Reduction

The linearised model was found to be non-minimal, and a number of techniques were considered to convert the model to a minimal system. In the model reduction technique used, the least observable and controllable states were discarded. Fortunately, the discarded uncontrollable states were stable. If this were not the case, these states would have grown indefinitely and a control solution for the system would not have been possible altogether.

It was also demonstrated that model reduction did not change the process behaviour appreciably for our frequency region of interest. Most model reduction techniques are based on transforming the state-space to another equivalent realisation with similar input-output characteristics. As a result, the biological significance of the states is lost. If the state information is of importance, this may pose a significant problem as there is no alternative method of obtaining a minimal realisation. In the context of insulin regulation, having the states retain their biological information is, although desirable, not essential.

6.1.3 Controller Design

Optimal control is well-suited for glucose regulation because of the fact that it can optimise insulin usage whilst still minimising time taken for states to reach nominal operating points. These desirable characteristics were conclusively demonstrated by analysing the closed-loop responses in addition to the control effort required.

To allow for comparison, a pole placement technique was also used. This technique considers the input/output relationship in tuning the controller parameters, and allows the dynamics of the glucose concentration to be modified as required to stay within physiological limits.

6.1.4 Observer Design

Having considered all these issues, we attempted to implement the Darouach functional observer. We found that modification of the feedback gain matrix was needed to satisfy the rank conditions required for the existence of the observer. This resulted in an observer that is of lower order than a reduced-order observer. Other techniques should be trialled to determine whether a different approach could yield an even lower order observer. However, the Darouach observer is stable, and its error asymptotically approaches zero making it a plausible design choice.

6.2 Future Work

There are a number of issues that are worthy of further attention. These issues are largely trivial and could be studied by using the techniques documented in this dissertation. These issues included (i) linearisation about different operating points, (ii) choosing the optimal order for model reduction, (iii) investigating further into pole locations and LQR matrix Q and R iterations, and (iv) investigating into different functional observer design techniques that may yield lower order observers for this application.

Being the primary focus of the dissertation, further work could be conducted in relation to the functional observer. Firstly, although the existence conditions for the Darouach observer were successfully relaxed, producing an observer of order less than $n - m$, the claimed observer of 'much less order' was not conclusively demonstrated. This was due to the existence conditions, initially not being satisfied. Future work could seek to obtain a different R matrix to append, further minimising the order. Most importantly however, the successful implementation of the functional observer into an application should pave the way for other developments such as implementing simultaneous functional observers.

As Moreno [42] notes, the simultaneous observation problem arises when a single observer is required to observe two or more linear plants; for example when a robust observer is designed to converge despite certain failures in components of the system. The ability of this observer to correctly estimate the states of a system which has two or more operating points, makes this technique particularly useful for the blood glucose regulation problem.

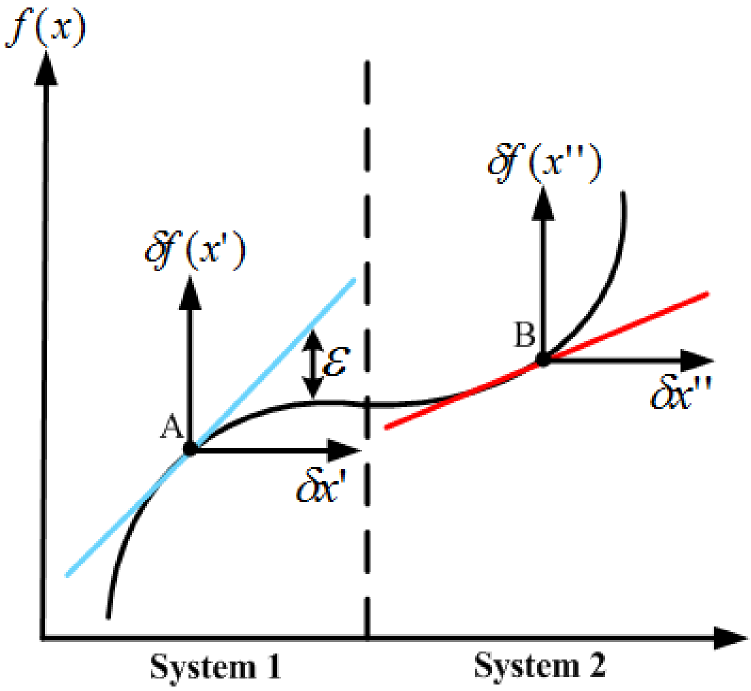

Figure 6.1: A Simultaneous Functional Observer Operating about Points A and B

If we consider Figure 6.1, the linearised approximation error $\mathcal{E}$ increases as the system operates away from the nominal operating point. This results in model error, which flows onto errors in control and observation. The simultaneous functional observer will give designers the option of designing for more than one operating point. As the number of operating points increase, the resolution of estimation improves, which could eventually yield a result for non-linear function of state estimation.

References

[1] D. Luenberger, "Observers for multivariate systems," *IEEE Transactions*, vol. AC-11, pp. 190–197, Apr. 1966.

[2] D. Luenberger, "An introduction to observers," *IEEE Transactions on Automatic Control*, vol. AC-16, pp. 596–602, Dec. 1971.

[3] M. Oishi. Lecture Notes - 29, 2006. University of British Columbia - EECE 360.

[4] G. Ellis, *Observers in Control Systems - A Practical Guide.* San Diego: Academic Press - Elsevier Science, 2002.

[5] K. Ogata, *Modern Control Engineering.* Prentice Hall, 3rd ed., 1997.

[6] R. Kalman and J. Bertram, "Control system analysis and design via the 'second method' of lyapunov-i. continuous-time systems," *Transactions ASME, ser.D,J. Basic Engineering*, vol. 82, pp. 371–393, 1960.

[7] R. Kalman, "Mathematical description of linear dynamical systems," *Journal SIAM, ser. A, on Control*, vol. 1, no. 2, pp. 152–192, 1963.

[8] F. F. R.D. Gupta and T. Hinamoto, "A direct procedure for the design of single functional observers," *IEEE Transactions on Circuits and Systems*, vol. CAS-28, no. 4, pp. 294–300, 1981.

[9] D. Luenberger, "Observing the state of a linear system," *IEEE Transactions on Military Electronics*, vol. MIL-8, pp. 74–80, 1964.

[10] M. Aoki and J. Huddle, "Estimation of the state vector of a linear stochastic system with a constrained estimator," *IEEE Transactions on Automatic Control*, pp. 432–433, 1967.

[11] F. Dellon and P. Sarachik, "Optimal control of unstable linear plants with inaccessible states," *IEEE Transactions on Automatic Control*, vol. AC-13, no. 5, pp. 491–495, 1968.

[12] E. Tse and M. Athans, "Optimal minimal-order observer-estimators for discrete linear time-varying systems," *IEEE Transactions on Automatic Control*, vol. AC-15, no. 4, pp. 416–426, 1970.

[13] R. Bass and I. Gura, "High-order system design via state-space considerations," (Atlanta, Georgia), Preprints Joint Automatic Control Conf., Jun. 1965.

[14] J. O'Reilly, *Observers for Linear Systems.* Academic Press, 1983.

[15] C. Tsui, "An algorithm for the design of single functional observers," *IEEE Transactions on Automatic Control*, vol. CAC-32, pp. 100–102, Jan 1985.

[16] M. Darouach, "Existence and design of functional observers for linear systems," *IEEE Transactions on Automatic Control*, vol. 45, pp. 940–943, May 2000.

[17] T. Fortmann and D. Williamson, "Design of low-order observers for linear feedback control laws," *IEEE Transactions on Automatic Control*, vol. AC-17, pp. 301–308, Jun. 1972.

[18] P. Murdoch, "Observer design for a linear functional of the state vector," *IEEE Transactions on Automatic Control*, vol. 18, pp. 308–310, Jun. 1973.

[19] J. Moore and G. Ledwich, "Minimal order observers for estimating linear functions of a state vector," *IEEE Transactions on Automatic Control*, vol. AC-20, no. 5, pp. 623–631, 1970.

[20] F. Fairman, "Hybrid estimators for discrete time stochastic systems," *IEEE Transactions*, vol. SMC-8, no. 12, pp. 849–854, 1978.

[21] M. Missaghie and F. Fairman, "Sensitivity reducing observers for optimal feedback control," *IEEE Transactions on Automatic Control*, vol. AC-11, pp. 952–957, 1977.

[22] C. Tsui, "A new algorithm for the design of multifunctional observers," *IEEE Transactions on Automatic Control*, vol. AC-30, pp. 89–93, Jan 1985.

[23] M. Aldeen and H. Trinh, "Reduced-order linear functional observer for linear systems," *IEEE Process-Control Theory Applied*, vol. 146, no. 5, pp. 399–405, 1999.

[24] P. Murdoch, "Design of degenerate observers," *IEEE Transactions on Automatic Control*, vol. 19, pp. 441–442, Aug. 1974.

[25] C. Tsui, "On the order reduction of linear function observers," *IEEE Transactions on Automatic Control*, vol. 31, pp. 447–449, May 1986.

[26] B. Datta and D. Sarkissian, "Block algorithms for state estimation and functional observers," in *Proceedings of the 2000 IEEE International Symposium on Computer-Aided Control System Design, Anchorage*, pp. 19–23, IEEE, Sep. 2000.

[27] P. V. Dooren, "Reduced order observers: A new algorithm and proof," *Elsevier*, vol. 4, no. 5, pp. 243–251, 1984.

[28] H. K. Khalil, *Nonlinear Systems*. Prentice Hall, 3rd ed., 2002.

[29] P. F. R.E. Kalman and M. Arbib, *Topics in Mathematical System Theory*. McGraw-Hill, 1969.

[30] L. Westphal, *Sourcebook of Control Systems Engineering*. Chapman and Hall, 1995.

[31] T. Kailath, *Linear Systems*. Prentice Hall Information and System Sciences Series, 1980.

[32] B. Moore, "Principal component analysis in linear system: Controllability, observability and model reduction," *IEEE Transactions on Automatic Control*, vol. AC26, p. 1732, 1981.

[33] N. S. Nise, *Control Systems Engineering*. John Wiley and Sons, 3rd ed., 2000.

[34] W. Wonham, "On pole assignment in multi-input controllable linear systems," *IEEE Transactions on Automatic Control*, vol. AC-12, pp. 660–665, 1967.

[35] A. Locatelli, *Optimal Control: An Introduction.* Birkhauser Verlag, 2001.

[36] F. J. D. Robert S. Parker and N. Peppas, "The intravenous route to blood glucose control," *IEEE Engineering in Medicine and Biology*, pp. 65–73, Jan. 2001.

[37] R. F. E. Ruiz-Valazquez and D. Campos-Delgado, "Blood glucose control for type 1 diabetes mellitus: A robust tracking h-infinity problem," *Elsevier - Control Eng. Practice*, no. 12, pp. 1179–1195, 2004.

[38] J. Sorenson, *A Physiologic Model of Glucose Metabolism in Man and its Use to Design and Assess Improved Insulin Therapies for Diabetes.* PhD thesis, Department of Chemical Engineering, MIT, 1985.

[39] G. B. F.C. Erzen and A. Cinar, "Simulation studies on the dynamics of diabetes mellitus," in *Proceedings of the IEEE international symposium on bio-informatics and biomedical engineering*, pp. 231–235, 2000.

[40] T. Deutsch and E. Lehmann, "A physiological model of glucoseinsulin interaction in type 1 diabetes mellitus," *Journal of Biomedical Engineering*, vol. 14, pp. 235–242, May. 1992.

[41] N. A. P. Robert S. Parker, Francis J. Doyle and J. H. Ward, "Robust h-infinity glucose control in diabetes using a physiological model," *AICHE Journal*, vol. 46, pp. 2537–2549, Dec. 2000.

[42] J. Moreno, "Simultaneous observation of linear systems: A state-space interpretation," *IEEE Transactions on Automatic Control*, vol. 50, pp. 1021–1025, Jul. 2005.

Appendix A

University of Southern Queensland
FACULTY OF ENGINEERING AND SURVEYING
ENG 4111/4112 Research Project
PROJECT SPECIFICATION

FOR: Marina Mendis

TOPIC: **APPLICATION OF A FUNCTIONAL OBSERVER FOR GLUCOSE REGULATION PROBLEM**

SUPERVISOR: Dr Paul Wen

ENROLMENT: ENG4111- S1 Ext, 2008;
ENG4112- S2 On Campus, 2008

PROJECT AIM: This project seeks to analyze a 19th Order insulin-glucose physiological model, linearise the system, apply a model reduction technique and design a state feedback controller which utilize Darouach's functional observer to supply the linear function of the states for feedback.

PROGRAMME: Issue A, 25th March 2008

(1) Do a literature review to investigate the current situation in this area
(2) Search for various approaches to solving the observer problem
(3) Provide a background to mathematical and theoretical concepts required in analysis and implementation of the functional observer.
(4) Introduce the full-state observer and the reduced-order observer and the construction of each
(5) Detail analysis of the Darouach's functional observer
(6) Provide an application for the functional observer. A glucose-insulin regulation problem using pole placement and implementation using Mat Lab
(7) Design desired output or input which entails linearization, model reduction through balanced truncation technique, control law design and finally a functional observer implemented to provide the linear state back.
(8) Develop controller to achieve the desired output.
(9) Simulate system (Model + Controller)
(10) Conclude the paper, summarizing the results and provide some suggestion for further areas of research.

As time permits:
(11) Design an improved controller based on simulation results.

AGREED:

.... (Student) .... (Supervisor)

Date: / /2008 Date: / /2008
Examiner/Co-examiner:.........................

Appendix B

Matlab Code

It should be noted that code has been modified for presentation. Matlab code should be run off the CD instead of copying from this section. Copying code from this section verbatim will not compile.

B.1 Linearisation of 19^{th} Order Non-linear Glucose-Insulin Model

```
clear
%Initialise constants
v_bC=3.5;     %dL
v_bT=4.5;     %dL
v_hC=13.8;    %dL
v_sC=11.2;    %dL
v_lC=25.1;    %dL
v_kC=6.6;     %dL
v_pC=10.4;    %dL
v_pT=67.4;    %dL
V_bC=0.265;  %L
V_hC=0.985;  %L
V_sC=0.945;  %L
V_lC=1.14;    %L
V_kC=0.505;  %L
V_pC=0.735;  %L
V_pT=6.3;     %L
V_n=9.93;     %L
q_b=5.9;      %dL/min
q_h=43.7;     %dL/min
q_s=10.1;     %dL/min
```

```
q_l=12.6;     %dL/min
q_a=2.5;      %dL/min
q_k=10.1;     %dL/min
q_p=15.1;     %dL/min
Q_b=0.45;     %L/min
Q_h=3.12;     %L/min
Q_s=0.72;     %L/min
Q_l=0.9;      %L/min
Q_a=0.18;     %L/min
Q_k=0.72;     %L/min
Q_p=1.05;     %L/min
F_pnc=0.910;  %L/min
T_b=2.1;      %min
T_pG=5.0;     %min
T_pI=20;      %min
F_kC=0.3;
F_lC = 0.4;
F_pC = 0.15;
Gamma_bu=70; %constant
Gamma_rbcu=10; %constant
Gamma_su=20; %constant
Gamma_meal=166.667;
Gamma_kc=13.161;
Gamma_ke=0;
Gamma_lc=28.515;
Gamma_pc=8.3239;
Gamma_pgu=97.79;
Gamma_pir=0; %No pancreatic insulin release
Gamma_pnr=3.1776;
Gamma_sia=0; %no insulin given at subcut depot
Gamma_ivi=-50;

%Initialise variables
syms G_bC G_bT G_hC G_sC G_lC G_kC G_pC G_pT A_ihgp A_nhgp A_ihgu
     I_bC I_hC I_sC I_lC I_kC I_pC I_pT N


f = [(G_hC-G_bC)*q_b/v_bC-(G_bC-G_bT)*(v_bT/(T_b*v_bC));                %1
     (G_bC-G_bT)/T_b-Gamma_bu/v_bT;                                      %2
     (G_bC*q_b+G_lC*q_l+G_kC*q_k+G_pC*q_p-G_hC*q_h-Gamma_rbcu)/v_hC;     %3
```

```
    (G_hC-G_sC)*q_s/v_sC+Gamma_meal/v_sC-Gamma_su/v_sC;                  %4
    (G_hC*q_a+G_sC*q_s-G_lC*q_l)/v_lC+(155*A_ihgp*[2.7*tanh(0.388*N)-
A_nhgp]*[1.425-1.406*tanh(0.6199*(G_lC/101-0.4969))])/v_lC-(20*
A_ihgu*[5.6648+5.6589*tanh(2.4375*(G_lC/101-1.48))])/v_lC;        %5
%Note: For our operating pt. G_kC<460mg/dL
    (G_hC-G_kC)*q_k/v_kC-(71+71*tanh(0.011*(G_kC-460)))/v_kC;            %6
    (G_hC-G_pC)*q_p/v_pC+(G_pT-G_pC)*v_pT/(T_pG*v_pC);                   %7
    (G_pC-G_pT)/T_pG-((35*G_pT)/86.81*[7.035+6.51623*tanh(0.33827*
       (I_pT/5.304-5.82113))])/v_pT;                                     %8
    1/25*[1.2088-1.138*tanh(1.669*I_lC/21.43-0.8885)-A_ihgp];            %9
    1/65*[(2.7*tanh(0.388*N)-1)/2-A_nhgp];                               %10
    1/25*(2*tanh(0.549*I_lC/21.43)-A_ihgu);                              %11
    (I_hC-I_bC)*Q_b/V_bC;                                                %12
    (I_bC*Q_b+I_lC*Q_l+I_kC*Q_k+I_pC*Q_p-I_hC*Q_h-Gamma_ivi)/V_hC;       %13
    (I_hC-I_sC)*Q_s/V_sC;                                                %14
    (I_hC*Q_a+I_sC*Q_s-I_lC*Q_l)/V_lC+(Gamma_pir-
     (F_lC*(I_hC*Q_a+I_sC*Q_s+Gamma_pir)))/V_lC  ;         %15
    (I_hC-I_kC)*Q_k/V_kC-(F_kC*I_kC*Q_k)/V_kC;                           %16
    (I_hC-I_pC)*Q_p/V_pC-(I_pC-I_pT)*V_pT/(T_pI*V_pC);                   %17
    (I_pC-I_pT)/T_pI+(Gamma_sia-I_pT/((1-F_pC)/F_pC*1/Q_p-1/
    (T_pI*V_pT)))/V_pT;                                      %18
    ([1.3102-0.61016*tanh(1.0571*(I_hC/15.15-0.46981))]*[2.9285-
2.095*tanh(4.18*(G_hC/91.89-0.6191))]-N)*F_pnc/V_n              %19

    ];

v = [G_bC, G_bT, G_hC, G_sC, G_lC, G_kC, G_pC, G_pT, A_ihgp,
A_nhgp, A_ihgu, I_bC, I_hC, I_sC, I_lC, I_kC, I_pC, I_pT, N];


J = jacobian(f,v)
```

B.2 Evaluate Jacobian at Nominal Operating Point

```
clear all
% A System matrix : Jacobian J evaluated at nominal operating point.
% Define nominal operating points
I_lC=19.59979141104307;
I_hC=32.66631901840506;
I_pT=11.43321165644186;
G_hC=87.64475678894864;
G_lC=97.92471015319093;
G_kC=87.64086499938452;
G_pT=80.43977141513683;
A_ihgp=1.15926668712183;
A_nhgp=-0.09965111223031;
A_ihgu=0.92755488089528;
N=0.78798420699630;
V_hC=0.985;  %L
A=[  ... 19x19 matrix
B=[0;
   0;
   0;
   0;
   0;
   0;
   0;
   0;
   0;
   0;
   0;
   0;
   -1/V_hC;
   0;
   0;
   0;
   0;
   0;
   0];
C=[0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0];
D=0;
rank(A)
```

B.3 Model Reduction

```
clear
% A System matrix : Jacobian J evaluated at nominal operating point.
% Define nominal operating points
I_lC=19.59979141104307;
I_hC=32.66631901840506;
I_pT=11.43321165644186;
G_hC=87.64475678894864;
G_lC=97.92471015319093;
G_kC=87.64086499938452;
G_pT=80.43977141513683;
A_ihgp=1.15926668712183;
A_nhgp=-0.09965111223031;
A_ihgu=0.92755488089528;
N=0.78798420699630;
V_hC=0.985;  %L
A=[ ... 19x19 matrix
B=[0;
   0;
   0;
   0;
   0;
   0;
   0;
   0;
   0;
   0;
   0;
   0;
   -1/V_hC;
   0;
   0;
   0;
   0;
   0;
   0];
C=[0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0];
D=0;
sys=ss(A,B,C,D);
T1start=[];
```

```
n=rank(sys.a);
fprintf( ...
'The order of System Matrix A is %g.\n',n)
r =rank(ctrb(sys)); %Since r<n => Must obtain minimal realisation
%Create T1 to isolate controllable states
for i=1:r
    T1start=[T1start A^(i-1)*B];
end
v1=[rand(n,n-r)];
T1=[T1start v1];
%Hard Code first T1 generated
T1=[0,0,0,0,0.004910168127286,-0.043523391454114,...,...];
rank(T1);
Abar=inv(T1)*A*T1;
Bbar=inv(T1)*B;
Cbar=C*T1;
%Setting a tolerance value for numbers less than 1E-7
for i=1:n,
    for j=1:n,
        if abs(Abar(i,j))<1E-7,
            Abar(i,j)=0;
        end
    end
end
% From inspection of Abar we decide that the cutoff
% should be 11th order
% Create controllable matrices: Ac, Bc, Cc
Ac = Abar(1:11,1:11);
Bc = Bbar(1:11);
Cc = Cbar(1:11);
%Use eig(Anotc) to check if non-controllable states are stable
Anotc=Abar(10:19,10:19);
fprintf( ...
'The controllability matrix of Ac,Bc is rank %g.\n',(rank(ctrb(Ac,Bc))))
fprintf( ...
'The observability matrix of Ac,Cc is rank %g.\n',(rank(obsv(Ac,Cc))))
n=rank(ctrb(Ac,Bc));
r=rank(obsv(Ac,Cc));
T2start=[];
for i=1:r
```

```
    T2start=[T2start;
             Cc*Ac^(i-1);];
end
v2=[rand(n-r,n)];
T2=[T2start;
    v2];
%Hard Code first T2 generated
T2=[0,0,0,0.004136174746978,-0.021976482959224,0.102551693023201,...,...];
rank(T2)
Acbar=T2*Ac*inv(T2);
Bcbar=T2*Bc;
Ccbar=Cc*inv(T2);
%Clean up matrix Acbar
%Setting a tolerance value for numbers less than 1E-7
for i=1:n,
    for j=1:n,
        if abs(Acbar(i,j))<1E-7,
            Acbar(i,j)=0;
        end
    end
end
% From inspection of Acbar we decide that the cutoff should be 9th order
% Create controllable observable matrices: Aco, Bco, Cco
Aco = Acbar(1:9,1:9);
Bco = Bcbar(1:9);
Cco = Ccbar(1:9);
fprintf( ...
'The controllability matrix of Aco,Bco is rank %g.\n',(rank(ctrb(Aco,Bco))))
fprintf( ...
'The observability matrix of Aco,Cco is rank %g.\n',(rank(obsv(Aco,Cco))))
% Perform balanced model reduction on 9th order system
[Ab,Bb,Cb,Db,totbnd,hsv]=balmr(A,B,C,D,1,9)
sysb=ss(Ab,Bb,Cb,Db);
[Y,T,X]=initial(sysb,[250*ones(1,1),zeros(1,8)],300);
%Vertical shift to account for 80.4 operating point
Y=Y+80.4*ones(length(Y),1);
plot(T,Y)
xlabel('Time (mins)');
ylabel('Glucose Concentration (mg/dL)');
```

B.4 Controller and Darouach Observer Design

```
clear all
Ab=[9x9 matrix];
Bb=[9x1 matrix];
Cb=[1x9 matrix];
Db=[0];
sysb=ss(Ab,Bb,Cb,Db);

%Controller Design
%*****************************************************************
%Poles of closed loop system scheme 1
%poles = eig(sysb)-[zeros(5,1);0.02;0;0;0];
%Poles of closed loop system scheme 2
poles = eig(sysb)-[zeros(6,1);0.05;0.05;0.05];
Lcont = place(Ab,Bb,poles);
sysnew=ss(Ab-Bb*Lcont,Bb,Cb,Db);
[Y,T,X]=initial(sysnew,[300*ones(1,1),zeros(1,8)],300);
%Vertical shift to account for 80.4 operating point
Y=Y+80.4*ones(length(Y),1);
[U]=Lcont*X';
%Vertical shift to account for 22.3 operating point
U=U+22.3*ones(1,length(U));
figure(1)
plot(T,Y);
grid
xlabel('Time (mins)');
ylabel('Glucose Concentration (mg/dL)');
figure(2)
plot(T,U);
grid
xlabel('Time (mins)');
ylabel('Insulin Control Effort (mU/min)');
A=Ab;
B=Bb;
C=Cb;
D=Db;
%Contruct new Lbar to relax existence condition
R= null([C*A;Lcont;C])';
L=[Lcont;R];
```

```
%Darouach Functional Observer
%*********************************************************************
%Test for Lemma 1 Equation (10)
rank([L*A;C*A;C;L])
rank([C*A;C;L])
if(rank([L*A;C*A;C;L]) ~= rank([C*A;C;L]))
    disp('Lemma 1 condition not met')
end;
%Define A_hat and C_hat from Equation (9)
A_hat = A*(eye(length(A))-pinv(L)*L);
C_hat = C*(eye(length(C))-pinv(L)*L);
%Define Sigma
Sigma = [C*A_hat; C_hat];

%Step 1) of Darouach's design algorithm: Obtain matrices F & G from
%Equations (18) and (19) respectively.

F = L*A*pinv(L)-L*A_hat*pinv(Sigma)*[C*A*pinv(L); C*pinv(L)]
G = (eye(length(Sigma*pinv(Sigma)))-Sigma*pinv(Sigma))
*[C*A*pinv(L); C*pinv(L)]
sizeg=size(G);
%Step 2)Z can be assigned arbitrarily
%Z=[zeros(sizeg(2),sizeg(1)-1),ones(sizeg(2),1)]
Z=[ones(sizeg(2),sizeg(1))]
% Equation (17)
N = F-Z*G
%Step 3) of design algorithm : Determin E,K and J
temp = L*A_hat*pinv(Sigma) + Z*(eye(length(Sigma*pinv(Sigma)))
-Sigma*pinv(Sigma))
[tempr,tempc]=size(temp);
E = temp(:,1:tempc/2)
K = temp(:,tempc/2+1:tempc)

J = K+N*E
%Step 4)
P = L-E*C
H = P*B

%Coding the System Dynamics
%**********************************************************
```

```
%Initialise variables with arbitrary initial conditions
% n=length(A);
% o=length(N);
% x(:,1) = [ones(n,1)];
% w(:,1) = [ones(o,1)];
% u = [5];
%
% for i=1:18000
%     x_dot = A*x(:,i)+B*u;
%     x(:,i+1)=x(:,i)+0.01*x_dot;
%     temp1 = L*x(:,i);
%     temp2 = C*x(:,i);
%     z1(:,i) = temp1;
%     y1(:,i) = temp2(:);
%
%     w_dot = N*w(:,i)+J*y1(:,i)+H*u;
%     w(:,i+1) = w(:,i)+0.01*w_dot;
%     temp3 = w(:,i)+E*y1(:,i);
%     z_hat1(:,i) = temp3;
% end
% %Plot error function of functional observer e =z-z_hat
% error=(z1(1,:)-z_hat1(1,:));
% plot(error)
% xlabel('Time'); ylabel('Error');

%Using Matlab in-built functions
%*******************************************************
%Define dimensions and the combined system matrix
m=length(A);
o=length(N);
B=[B zeros(m,o-1)]
% Construct composite system
Acomb=[A-B*L,-B;zeros(o,m),N]
n=length(Acomb);
% Simulate to t secs
time=25;
sys=ss(Acomb,eye(n),eye(n),eye(n));
x0=[50*ones(m,1);zeros(o,1)]

%Simulate the time response
```

```
t=0:0.1:time;
[y,t,x]=initial(sys,x0,t);

e2=x';
e2=e2(n-o:n,:);

figure(3)
%Plot the time response
plot(t,e2(1,:),'-');grid %Observer Error
xlabel('Time (secs)'); ylabel('Error');
```

B.5 LQR Controller Design

```
clear all
Ab=[9x9 matrix];
Bb=[9x1 matrix];
Cb=[1X9 matrix];
Db=[0];
sysb=ss(Ab,Bb,Cb,Db);
%LQR Controller Design
%******************************************************************
Q=Cb'*Cb;
R=input('Input R = ');              %Weightiing for control effort
[L]=lqr(sysb,Q,R)  % Feedback gain matrix
sysnew=ss(Ab-Bb*L,Bb,Cb,Db);
[Y,T,X]=initial(sysnew,[300*ones(1,1),zeros(1,8)],300);
%Vertical shift to account for 80.4 operating point
Y=Y+80.4*ones(length(Y),1);
[U]=L*X';
%Vertical shift to account for 22.3 operating point
U=U+22.3*ones(1,length(U));
figure(1)
plot(T,Y);
grid
xlabel('Time (mins)');
ylabel('Glucose Concentration (mg/dL)');
figure(2)
plot(T,U);
grid
xlabel('Time (mins)');
ylabel('Insulin Control Effort (mU/min)');
```