© 2010 Matthew Aloysius Ralph

$\mathscr{L}_1$-ADAPTIVE CONTROL FOR ANESTHESIA DELIVERY

BY

MATTHEW ALOYSIUS RALPH

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Systems & Entrepreneurial Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2010

Urbana, Illinois

Adviser:

Associate Professor Carolyn Beck

ABSTRACT

In this paper, we discuss the first application of recently developed $\mathscr{L}_1$-adaptive control methods for the closed-loop control of anesthesia delivery during surgery. Our initial objective, described herein, is to design controllers that are robust to inter-patient variability, such that patients follow a prespecified Bispectral Index (BIS) profile. The controllers are then evaluated on their ability to reject the disturbances effects on the patients' BIS trajectory that arise during the course of surgery. The controllers are designed using identification-based models constructed using clinical trial data. Two different $\mathscr{L}_1$-adaptive control schemes, a state-feedback scheme and an output-feedback scheme, will be evaluated and compared in terms of tracking performance, amount of anesthesia used, and robustness to interpatient variability.

To my wife Shannon, for her love and support.

ACKNOWLEDGMENTS

First, I would like to thank Professor Carolyn Beck for her support and guidance during my time in graduate school. Thanks to Dr. Marc Bloom Ph.D., M.D. who provided the data used in this research, his experience, and medical expertise, without which this research would not have been possible. I would also like to thank Evgeny Kharisov and Jan Vervoorst for additional insight and help into ways to further improve the performance. I would like to thank the National Science Foundation for providing their financial support for this research.

Finally, I would like to give special thanks to my family and friends. To my parents for supporting me and giving me the opportunities that have let me accomplish all that I have. To Woody, Joe, Steph, Mike, and John for standing by me and lending me a sounding board when I needed one. Most of all, I would like to thank my wife Shannon for her love and support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

During surgery, the anesthesiologist continuously monitors and adjusts the delivery of anesthesia to the patient in an attempt to maintain a desired level of *sedation, analgesia* and *muscle relaxation*. Concurrently, the anesthesiologist maintains patient ventilation parameters and monitors cardiovascular and respiratory functions such as heart rate (HR), blood pressure (BP), oxygen saturation and end-tidal (exhaled) carbon dioxide (CO_2) levels. Invasive montitoring is sometimes used by attending anesthesiologists to directly measure not only arterial blood pressure (ABP), but right-heart filling pressures (CVP), left-heart filling pressures (PCWP), and pulmonary arterial pressures. Cardiac output (CO) may be measured by thermodilation methods and then used to derive systemic vascular resistance (SVR), pulmonary vascular resistance (PVR) and a host of other cardiac performance measures. Additionally, intra-operative blood samples are often taken and used by the anesthesiologist to observe gas concentrations, blood-sugar levels, electrolyte concentrations and coagulation parameters. In short, the anesthesiologist performs an extremely complex role, namely, that of a multivariable feedback controller. A common long term research goal in this area is thus to incorporate partially automated anesthesia delivery into the process, allowing the anesthesiologist to concentrate on urgent safety-critical events that arise during surgery.

In order to implement model-based feedback control of anesthesia delivery, two primary needs are (1) adequate and appropriate means of sensing the patient's level of sedation, analgesia and muscle relaxation, and (2) mathematical models capturing the patient response to anesthetic agents. Over the past two decades, the *bispectral index* (BIS), a statistical index based on phase and frequency relations between the component frequencies in EEG recordings, has found significant acceptance as a measure of sedation level (see [1, 2, 3] and the references therein). The adequacy and extent of muscle relaxation or *neuromuscular blockade* (NMB) can be evaluated effectively using different modes of electrical stimulation: single-twitch mode (i.e., a single supramaximal electrical stimulus pulse)

is useful after sedative agents have been administered to the patient but prior to the administration of muscle relaxants; train-of-four (TOF) modes (i.e., four spra-maximal electrical stimuli) are useful to determine readiness for intubation and throughout surgery to evaluate proper maintanence of adequate NMB levels; and TOF or double-burst stimulation modes are useful for assessing recovery from NMB agents. Surface electrodes and/or piezoelectric sensors are used to measure patient response to such electrical stimuli [4, 5]. Whereas the means and methods noted above for establishing and monitoring adequate sedation and NMB levels are by now fairly well-accepted, there is no standardized or generally accepted method for determining the state of analgesia at this time. Under the administration of sedatives, analgesia is continually influenced by external stimuli and the administration of analgesic drugs, and interaction, or synergy, between analgesics and sedatives is, for the most part, unavoidable. Some common anesthetic drugs are propofol, isoflurane, fentanyl, and remifentanil. In our studies, the anesthetic agent is the drug isoflurane. While sedated, the patient experiences a variety of external stimuli that affect his or her vitals. These include incisions, insertion of the laryngal mask, jostling of the patient, and sutures. Autonomic reactions, such as tachycardia, hypertension, sweating and lacrimation, are typically considered signs of inadequate analgesia. Autonomic monitoring techniques, such as the analysis of heart rate variability, laser Doppler flowmetry, phlethysmographically derived indices (i.e., pinching in the webbing between the fingers) and the pupillary light reflex, may help to quantitate reactions of the autonomic nervous system, but no verified sensing-metric combination has yet been established. In this paper, the main focus of our discussion will be on controlling the *level of sedation* via automated feedback methods, in particular implementing novel $\mathscr{L}_1$-adaptive control techniques. We want the anesthesiologist to be able to set a BIS reference trajectory for the patient while undergoing surgery and have the controller maintain that sedation level automatically.

Most of our studies have utilized clinical trial data where the anesthesiologist used isoflurane and the disturbances the patient experienced were laryngal mask insertion and removal (LMA), squeezing of the trapezius and yelling at the patient (EVAL), and small electrical stimulation (EP). Our long term goals include extending these methods to the mutli-input multi-output (MIMO) case described above.

Our initial objective, described herein, is to design a controller such that patients under anesthesia follow a prespecified BIS profile and, simultaneously, cer-

2

tain vital signs are maintained in a safe range, ensuring that the proper level of sedation is maintained throughout the surgery without putting the patient at risk. The BIS value is a single dimensionless number ranging from 0 to 100, where 0 corresponds to a silent EEG, and 100 corresponds to a patient being fully awake and aware. A BIS value between 60 and 40 is considered a viable level for general anesthesia, where the patient is not aware and surgery can be performed [6]. The manner in which the BIS level of any patient responds to the infusion and/or inspiration of anesthetic agents is not linear. In fact, the standard modeling paradigm that commonly has been used to describe the relationships between anesthetic inputs and patient outputs (or effects) is that of *compartment models*, which consist of a linear time-invariant (LTI) system cascaded with a static nonlinearity.

More precisely, pharmacokinetic (PK) compartment models are widely used as a means of predicting the disposition of drug in the body, by modeling the simultaneous diffusion of drug through body tissues and the flow of drug in blood. Most drugs are characterized by PK models containing a *central compartment*, which typically has a drug concentration corresponding to that of the blood, and peripheral compartments that represent groupings of internal organs and tissues of the body; these models form the LTI system. A theoretical *effect compartment* is then added to capture the concentration-to-effect relationship. The effect compartment typically consists of a static nonlinearity plus an additional first order LTI system that reflects the time-lag in the patient response to anesthesia, and is frequently referred to as a pharmacodynamic (PD) model (see [7, 8, 9, 10, 11] for details). The resulting (*grey box*) mathematical models are inherently single-input single-output (SISO) and consist of a system of ordinary differential equations plus a nonlinear function, representing the relations between the drug input function, the concentration of drug in the various compartments, and the effect of the drug on specific patient endpoints. As these models are strictly SISO, they are incapable of capturing the effects of disturbances, drug synergies, or coupling among effects in the human body.

Alternatively, in more recent previous work it has been shown that patient response to anesthesia can be adequately captured by *multivariable* piecewise-linear models, with one linear model capturing patient response around an "awake" equilibrium state, and another linear model capturing patient response around a "sedated" equilibrium state [12, 13]. These models incorporate external stimuli inputs and certain vital sign outputs. In our current work, we derive a set of models that correspond to patient response to anesthesia and external stimuli in an

3

"awake" state, and in a "sedated" state, respectively, for seven different patients' sets of data. Using these models, in this thesis we discuss control synthesis and simulation results using the recent $\mathscr{L}_1$-adaptive control methods proposed in [14] and [15], which adapt quickly, ensure stable performance in the presence of model uncertainties, achieve the desired BIS reference tracking objectives, and are robust to interpatient variability.

In this thesis, we present the first application of $\mathscr{L}_1$-adaptive control techniques for closed-loop control of anesthesia delivery during surgery. We investigate both state and output feedback schemes. With these controllers, we are able to achieve small tracking error while maintaining robustness with respect to the variability found across different patient responses to anesthesia. With the output feedback scheme, depending on tuning, we have found that one controller can be used on many patients, without any design modifications being made to it. Additionally, the output-feedback controller compensates for disturbances, like incisions and intubation, the patient may experience, keeping the patient sedated throughout the procedure.

1.1 Prior work

A number of prior control efforts for the anesthesia problem have been completed and evaluated over the past 50 years. Schwilden and colleagues used median frequencies from EEG power spectra as one measure of sedative effect to develop PK-PD model-based adaptive feedback control of the anesthetic agents propofol, methohexital, and alfentanil during both clinical studies and for surgery [16, 17, 18]. A number of model-based closed-loop anesthesia control studies also have been published by Gentilini and colleagues [19, 20, 21, 22, 23]. In [19], semi-physiological models and rule-based controllers for the regulation of respiratory functions and mean arterial pressure (MAP) under administration of isoflurane are described. The application of model predictive control schemes to regulate MAP during delivery of isoflurane is investigated in [20]. In one of the most comprehensive control implementations completed to date, Gentilini *et al.* proposed a control scheme for the regulation of MAP and sedation level using PK-PD models for the anesthetic agent isoflurane [21, 22], in which the design of a cascaded internal model control (IMC) controller to regulate the sedative effects of anesthesia via the BIS level of the patient, and a three-observer-based

4

state feedback controller to regulate MAP are proposed; the control designs are implemented in a loop-at-a-time manner. Mortier also considered control of sedation level via BIS monitoring in [24], where PK-PD model-based adaptive control of propofol is implemented in surgeries. More recently, Haddad, Hayakawa and Bailey have completed adaptive and neural network based control designs for the regulation of unconciousness under administration of propofol [25, 26]. Although this by no means represents an exhaustive discussion of prior work on closed-loop control of anesthesia, it presents the work most closely related to that discussed in this paper. However, all of the prior and ongoing work discussed above considers the use of SISO models and control designs, whereas our long term objective is to develop MIMO models and control designs.

The rest of the thesis is organized as follows.

Chapter 2: Problem Formulation

We introduce the basic model structures required for implementing the $\mathscr{L}_1$-adaptive control techniques. Namely, the state-space feedback architecture requires the system to be modeled as a SISO state-space model, and the output-feedback architecture requires the system to be modeled as a strictly proper (albeit unknown) transfer function.

Chapter 3: $\mathscr{L}_1$-Adaptive Control

The $\mathscr{L}_1$-adaptive control paradigm is a recently developed architecture that allows for fast adaptation while also being robust to uncertainty, disturbances, and, in our case, interpatient variability. In this chapter we provide an overview of the theoretical foundations for the $\mathscr{L}_1$-adaptive control framework. The fundamental $\mathscr{L}_1$-adaptive control's design is comprised of a predictor (state or output), a projection based adaptive law, an $\mathscr{L}_1$-stability condition, and a control law that incorporates a filter $C(s)$.

Chapter 4: Modeling and System Identification

In chapter 4, we outline the patient model development methods we have used. Data-based patient models are identified using subspace identification methods. The state-feedback scheme requires identification of both *black box* and *grey box* models to evaluate tracking and robustness to interpatient variability, respectively. For the output-feedback scheme, the *black box* models identified for use with the state-space scheme are sufficient for our studies, as this scheme is transfer function based.

Chapter 5: Simulation Results

Simulations have been completed for Patients 1, 2, 3, 5, 6, and 7. In this chap-

ter we present the results of these simulations and discuss the subsequent performance analysis. For each patient model a state-feedback and an output feedback controller have been designed and evaluated based on their ability to track a specific BIS trajectory and the total amount of anesthesia used. In addition, each controller also has been applied to all patient models to examine robustness to interpatient variability.

Chapter 6: Conclusions and Ongoing Research

Future directions such as higher order filters and real surgery condtions are discussed. Conclusions, observations, and comments are also addressed herein.

CHAPTER 2

PROBLEM FORMULATION

Our objective is to develop a control design platform using data-based models and advanced control techniques so that patients' BIS levels track a desired reference trajectory. Generally when going into surgery, the patient is lightly sedated by the anesthesiologist, which is approximately when the patient's BIS level is between 70 to 50. This BIS range, however, is not deeply sedated enough to perform surgery. To prevent the patient from waking up, the anesthesiologist wants to bring the patient into the heavily sedated region, a BIS value in the range 45-35. This range correlates with a sedation level for which any incisions or other invasive procedures can be performed. The patient is then slowly woken up by bringing them up to the lightly sedated region and finally into the alert region.

The way the human body reacts to stimuli varies greatly from person to person, and thus it is difficult to derive a single mathematical model that can be used for different people. As a result, the anesthesia-delivery controller must simultaneously compensate for the variability inherent in patient response to anesthesia and disturbances. Therefore, we require a control architecture that is both adaptive and robust. In order to achieve these objectives, we propose the use of the recently developed $\mathscr{L}_1$-adaptive control techniques described in [14] and [15]. These controllers are based on assuming models of specific forms with specific assumptions imposed upon them. We provide more details in this and the following chapter on the required model structures and assumptions.

2.1 $\mathscr{L}_1$-Adaptive Control - State Feedback Structure

The first $\mathscr{L}_1$-adaptive control scheme we will utilize is a state-feedback scheme described in [14]. To implement this method, we require that the patient's response, in awake and sedated states, is modeled in the following basic state-space

www.manaraa.com

form:

$$\begin{aligned}
\dot{x}(t) &= Ax(t) + Bu(t) \\
y(t) &= C^T x(t),
\end{aligned} \tag{2.1}$$

where $x \in \mathbb{R}^n$ is the state vector, u is the control signal (in our case, a percentage concentration, by volume, of the inhalational sedative isoflurane), y is the BIS output signal, $B, C \in \mathbb{R}^n$ are known constant vectors, and A is an unknown nxn Hurwitz matrix.

One crucial requirement of the state-feedback $\mathcal{L}_1$-adaptive controller architecture is the following assumption: there exists a Hurwitz nxn matrix A_m and a vector of parameters θ such that (A_m, B) is controllable and $A_m - A = B\theta^T$. Further, it is also assumed that the parameter vector θ resides in a known convex set Θ. The matrix A_m is chosen to give the system its desired behavior and θ is the vector of parameters that satisfies the previously discussed matching assumption relating the unknown matrix A with our chosen matrix A_m. If these assumptions are satisfied, we can rewrite all patient models in the following form:

$$\begin{aligned}
\dot{x}(t) &= A_m x(t) + B\left(u(t) - \theta^T x(t)\right), x(0) = x_0 \\
y(t) &= C^T x(t),
\end{aligned} \tag{2.2}$$

where $\theta \in \mathbb{R}^n$ is a vector of time-varying unknown parameters. In order to control and compensate for the varying responses to isoflurane among patients, we will design an adaptive controller to ensure that the patient's BIS level tracks a given reference signal $r(t)$ in both the transient and steady-state while keeping all of the remaining error signals bounded.

Remark 1. *We desire that the system in* (2.2) *have the same behavior as the desired reference system*

$$\begin{aligned}
\dot{x}_m(t) &= A_m x_m(t) + B k_g r(t), x_m(0) = x_0 \\
y_m(t) &= C^T x_m(t).
\end{aligned} \tag{2.3}$$

where $k_g = 1/(-C^T A_m^{-1} B)$. *This choice of* k_g *ensures zero steady-state error for constant inputs of* $r(t)$. *This desired behavior for* (2.2) *could be achieved if* $u(t)$ *in* (2.2) *is set to be* $u(t) = \theta^T x(t) + k_g r(t)$. *This controller cannot be directly implemented as* θ *is not known and must therefore be estimated.*

8

2.2 $\mathscr{L}_1$-Adaptive Control - Output Feedback Structure

The second scheme we will use is an output feedback controller, described in detail in [15]. We consider the following SISO system:

$$y(s) = A(s)(u(s) + d(s)) \tag{2.4}$$

where $y(t) \in \mathbb{R}$ is the system output (BIS value) and $u(t) \in \mathbb{R}$ is our control input (percent ISO). It is assumed that $A(s)$ is an unknown transfer function that is strictly proper. The $d(s)$ term is the Laplace transform of the time-varying disturbances $d(t)$ (see [15]). We have the following assumptions imposed on the disturbances $d(t) = f(t, y(t))$:

1. There exists constants $L > 0$ and $L_0 > 0$ such that $|f(t, y_1) - f(t, y_2)| \leq L|y_1 - y_2|$ and $|f(t, y)| \leq L|y| + L_0$ hold uniformly in $t \geq 0$.

2. There exist constants $L_1 > 0$, $L_2 > 0$, and $L_3 > 0$ such that

$$|\dot{d}(t)| \leq L_1|\dot{y}(t)| + L_2|y(t)| + L_3, \ \forall t \geq 0. \tag{2.5}$$

The control objective is to design an adaptive output feedback controller such that the patients' BIS level $y(t)$ tracks the BIS reference input following a desired reference model:

$$y(s) \approx M(s)r(s). \tag{2.6}$$

One example of this, and the form used later in the simulations is a first order system like

$$M(s) = m/(s+m), \ m > 0. \tag{2.7}$$

It can be seen that (2.4) can be re-written as

$$\begin{aligned} y(s) &= M(s)(u(s) + \sigma(s)) \\ \sigma(s) &= ((A(s) - M(s))u(s) + A(s)d(s))/M(s). \end{aligned} \tag{2.8}$$

Remark 2. *Note that if we could set $u(s) = r(s) - \sigma(s)$ in (2.4) we would achieve the desired behavior in (2.6). However, since $\sigma(s)$ is unknown, it must be estimated.*

Remarks 1 and 2 motivate the need for the $\mathscr{L}_1$-adaptive control designs detailed in Chapter 3.

9

CHAPTER 3

$\mathscr{L}_1$-ADAPTIVE CONTROL: BACKGROUND

The $\mathscr{L}_1$-adaptive control architecture was first proposed by Cao and Hovakimyan in [27, 14], and is intended for situations in which the controller must ensure that the system output follows a given reference signal in the face of modeling uncertainties. $\mathscr{L}_1$-adaptive control techniques have seen increasing application in the last few years, particularly in aerospace applications [28, 29]. These methods are useful in the same settings in which model reference adaptive control (MRAC) techniques could be considered, but also in settings in which *fast* adaptation is required. In this setting, MRAC controllers may result in instability and unacceptable behaviors, such as high frequency oscillations in the control channel and parameter drifts. However, $\mathscr{L}_1$-adaptive controllers prevent these behaviors from occurring based on the implementation of a projection operator in the adaptive law and a low pass filter in the control law. Additionally, the $\mathscr{L}_1$-adaptive control architecture guarantees *uniformly bounded* asymptotic and transient tracking for the system inputs and outputs. These bounds (which are quantifiable and discussed in the sequel) improve as the adaptive rate is increased.

3.1 Indirect vs. Direct Adaptive Control and relationship with $\mathscr{L}_1$-Adaptive Control

There are two different approaches to model reference adaptive control where the plant is unknown: indirect and direct control. The premise behind indirect adaptive control is to estimate the parameters of the unknown plant based on input and output data, and, based on these estimates, generate a feedback control function to update the parameters of the controller. That is, it estimates the true parameters of the plant at every time instant and uses it in the same non-adaptive control law. Direct control, conversely, does not explicitly try to identify the plant in order to generate the feedback control signal to update the parameters of the controller.

www.manaraa.com

That is, the parameters of the controller are directly adjusted. To do so, the system is first parameterized in terms of the controller parameters, and then, an adaptive law is designed to estimate the controller parameters. The objective of both approaches to the model reference adaptive control problem is to drive the difference between the output of the plant and the output of the reference model to zero [30].

In the state-feedback $\mathscr{L}_1$-adaptive control architecture we are estimating the constant parameter vector θ which is a parameter of the controller. Thus, it can viewed as a form of direct adaptive control. The output-feedback $\mathscr{L}_1$-adaptive control architecture estimates $\sigma(t)$ which is a controller parameter. That is, it also is a form of direct adaptive control.

What the $\mathscr{L}_1$-adaptive control scheme does is estimate an uncertain parameter of the plant (i.e. θ in the state-feedback architecture or σ in the output feedback architecture) and use that estimate in the control law to generate the control signal. Similar to MRAC, a desired reference model is selected that specifies the ideal response of the system to an external command. That is, the reference model should address required performance specifications such as percent overshoot, rise time, and settling time.

Where the $\mathscr{L}_1$-adaptive control approach differs from other adaptive schemes is that it is a control scheme that decouples adaptability and robustness. It offers the ability to have high adaptation rates while maintaining a time-delay margin that is bounded away form zero. This architecture guarantees fast adaptation that is limited only by hardware [31].

3.2 $\mathscr{L}_1$-Adaptive Control - State Feedback Based Design

The main elements of the $\mathscr{L}_1$-adaptive control paradigm are summarized below (see references [14] and [32] for details):

State Predictor: Consider the state predictor:

$$
\begin{aligned}
\dot{\hat{x}}(t) &= A_m\hat{x}(t) + B\left(u(t) - \hat{\theta}^T x(t)\right), \hat{x}(0) = \hat{x}_0 \\
\hat{y}(t) &= C^T\hat{x}(t),
\end{aligned}
\tag{3.1}
$$

where the adaptive estimate $\hat{\theta}(t)$ is governed by the adaptive law given in the following.

11
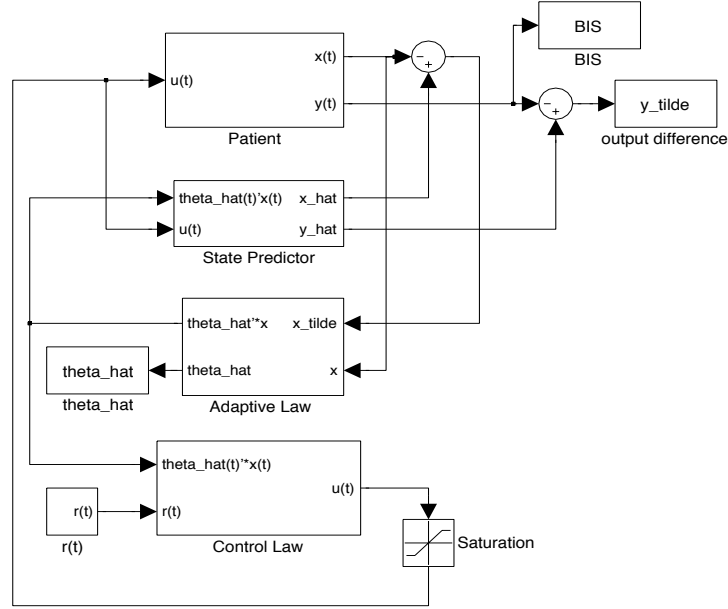
Figure 3.1: State feedback controller simulink model

Adaptive Law: The adaptive estimate $\hat{\theta}(t)$ is given by

$$\dot{\hat{\theta}}(t) = \Gamma \text{Proj}\left(\hat{\theta}(t), x(t)\tilde{x}(t)PB\right), \hat{\theta}(0) = \hat{\theta}_0, \tag{3.2}$$

where $\tilde{x}(t) = \hat{x}(t) - x(t)$ is the error signal between the states of the system and the state predictor, $\Gamma \in \mathbb{R}^+$ is an adaptive gain, and $P = P^T$ is the solution to the algebraic Lyapunov equation $A_m^T P + PA_m = -Q, Q > 0$. The projection operator $\text{Proj}(\cdot, \cdot)$ is essentially of least-squares form, and ensures that the parameter estimate $\hat{\theta}(t)$ remains inside the compact set Θ; see [32] for details.

Control Law: The control signal $u(t)$ is defined as the output of a low-pass filter. The control law is given by:

$$u(s) = C(s)\left(\bar{r}(s) + k_g r(s)\right), k_g = -1/\left(C^T A_m^{-1} B\right), \tag{3.3}$$

12

where $C(s)$ is a low-pass filter that is strictly proper and stable with DC gain $C(0) = 1$, and $\bar{r}(s)$ is the Laplace transform of $\bar{r}(t) = \hat{\theta}^T(t)x(t)$.

$\mathscr{L}_1$-gain Stability Condition: The filter $C(s)$ must be chosen such that the following condition is satisfied:

$$\|\bar{G}(s)\|_{\mathscr{L}_1}L < 1, \tag{3.4}$$

where $\bar{G}(s) = H(s)(1 - C(s)), H(s) = (s\mathbb{I} - A_m)^{-1}B$ and $L = \max_{\theta \in \Theta}\|\theta\|_{\mathscr{L}_1}$.

Transient Performance: The following upper bound holds for all $t \geq 0$ [14]:

$$\|\tilde{x}(t)\|_{\mathscr{L}_\infty} \leq \sqrt{\frac{\bar{\theta}_{\max}}{\lambda_{\min}(P)\Gamma}}, \bar{\theta}_{\max} = \max_{\theta \in \Theta}\sum_{i=1}^{n}4\theta_i^2, \forall t \geq 0 \tag{3.5}$$

The higher the adaptive gain value used, the lower the tracking error achieved.

Steady-State Performance: In the following two theorems, excerpted from [14], we summarize the guaranteed performance of the system when controlled by the $\mathscr{L}_1$-adaptive controller.

Theorem 1. *Given the system in* (2.2) *and an $\mathscr{L}_1$-adaptive controller as defined by* (3.1), (3.2), *and* (3.3) *subject to* (3.4), *the tracking error $\tilde{x}(t)$ converges asymptotically to zero.*

Theorem 2. *Given the system in* (2.2) *and an $\mathscr{L}_1$-adaptive controller defined by* (3.1), (3.2), *and* (3.3) *subject to* (3.4), *if $r(t)$ is a constant value r, then $y(t)$ converges to r.*

Using clinical trial data, we construct a set of data-based models in the form of (2.1). We then apply the $\mathscr{L}_1$-adaptive techniques outlined in the preceding, so that the patients' BIS levels track a prespecified reference trajectory.

3.3 $\mathscr{L}_1$-Adaptive Control - Output Feedback Based Design

Closed-loop Reference System: In [15] the following closed-loop reference system is considered:

$$y_{ref}(s) = M(s)(u_{ref}(s) + \sigma_{ref}(s)) \tag{3.6}$$

$$\sigma_{ref}(s) = \frac{(A(s) - M(s))u_{ref} + A(s)d_{ref}(s)}{M(s)} \tag{3.7}$$

13

$$u_{ref}(s) = C(s)(r(s) - \sigma_{ref}(s)) \tag{3.8}$$

$C(s)$ is a strictly proper transfer function with $C(0) = 1$. The simplest choice for $C(s)$ would be a first order system

$$C(s) = \omega/(s + \omega); \tag{3.9}$$

this is what is used in the majority of our simulations.

$\mathscr{L}_1$-**Stability Condition:** Choices of $M(s)$ and $C(s)$ are restricted such that

$$H(s) = \frac{A(s)M(s)}{(C(s)A(s) + (1 - C(s))M(s))} \tag{3.10}$$

is BIBO stable and

$$\|G(s)\|_{\mathscr{L}_1} L < 1, \ G(s) = H(s)(1 - C(s)). \tag{3.11}$$

If $M(s)$ and $C(s)$ are chosen such that (3.10) and (3.11) hold, then the discussed closed-loop reference system in (3.6), (3.7), and (3.8) is BIBO stable [15].

We choose $P > 0$ and define $Q = 2mP$. We then define

$$H_0(s) = \frac{A(s)}{C(s)A(s) + (1 - C(s))M(s)} \tag{3.12}$$

$$H_1(s) = \frac{(A(s) - M(s))C(s)}{C(s)A(s) + (1 - C(s))M(s)}. \tag{3.13}$$

If we define

$$A(s) = \frac{A_n(s)}{A_d(s)}, \ C(s) = \frac{C_n(s)}{C_d(s)}, \ M(s) = \frac{M_n(s)}{M_d(s)}, \tag{3.14}$$

then from (3.14) and (3.12) we have

$$H_0(s) = \frac{C_d(s)A_n(s)M_d(s)}{H_d(s)}, \tag{3.15}$$

and from (3.14) and (3.13) we have

$$H_1(s) = \frac{C_n(s)A_n(s)M_d(s) - C_n(s)A_d(s)M_n(s)}{H_d(s)} \tag{3.16}$$

where $H_d(s) = C_n(s)A_n(s)M_d(s) + M_n(s)A_d(s)(C_d(s) - C_n(s))$.

14

These expressions highlight the following: (1) since it can be seen that the difference in order between $C_d(s) - C_n(s)$ and $C_n(s)$ is larger than zero, the order of $M_n(s)A_d(s)(C_d(s) - C_n(s))$ must be higher than $C_n(s)A_d(s)M_n(s)$; (2) since it can be seen that the difference in order between $A_d(s)$ and $A_n(s)$ is larger than zero while the relative order of $M_n(s)$ and $M_d(s)$ is -1, it can be shown that the order of $M_n(s)A_d(S)(C_d(s) - C_n(s))$ must be higher than that of $C_n(s)A_n(s)M_d(s)$. Thus we know that $H_1(s)$ is strictly proper. Since $H_1(s)$ has the same denominator as $H(s)$, it is also bounded-input bounded-output (BIBO) stable. Similarly, it can be shown that $H_0(s)$ is both proper and BIBO stable.

Define

$$
\begin{aligned}
\Delta = & ||H_1(s)||_{\mathscr{L}_1}||r||_{\mathscr{L}_\infty} + ||H_0(s)||_{\mathscr{L}_1}(L\rho + L_0) \\
& + \left(\left\| \frac{H_1(s)}{M(s)} \right\|_{\mathscr{L}_1} + L||H_0(s)||_{\mathscr{L}_1} \frac{||C(s)H(s)/M(s)||_{\mathscr{L}_1}}{1 - ||G(s)||_{\mathscr{L}_1}L} \right)\bar{\gamma}
\end{aligned}
\tag{3.17}
$$

where $\bar{\gamma} < 0$ is an arbitrary constant. As was previously discussed, we have $H_1(s)$ strictly proper and BIBO stable, which implies $||H_1(s)/M(s)||_{\mathscr{L}_1}$ is finite. Thus Δ is a finite number. Further, define

$$
\beta_1 = 4\Delta||H_0(s)||_{\mathscr{L}_1}\left(L_1\beta_{01} + L_2 \frac{||C(s)H(s)/M(s)||_{\mathscr{L}_1}}{1 - ||G(s)||_{\mathscr{L}_1}L} \right),
\tag{3.18}
$$

$$
\beta_2 = 4\Delta||sH_1(s)||_{\mathscr{L}_1}(||r||_{\mathscr{L}_\infty} + 2\Delta) + 4\Delta||H_0(s)||_{\mathscr{L}_1}(L_1\beta_{02} + L_3 + \rho L_2),
\tag{3.19}
$$

$$
\beta_3 = P\beta_1/Q = \beta_1/(2m),
\tag{3.20}
$$

and

$$
\beta_4 = 4\Delta^2 + P\beta_2/Q = 4\Delta^2 + \beta_2/(2m)
\tag{3.21}
$$

where

$$
\rho = \frac{||H(s)C(s)||_{\mathscr{L}_1}||r||_{\mathscr{L}_\infty} + ||G(s)||_{\mathscr{L}_1}L_0}{1 - ||G(s)||_{\mathscr{L}_1}L},
\tag{3.22}
$$

$$
\beta_{01} = ||sH(s)(1 - C(s))||_{\mathscr{L}_1} \frac{L||C(s)H(s)/M(s)||_{\mathscr{L}_1}}{1 - ||G(s)||_{\mathscr{L}_1}L},
\tag{3.23}
$$

15

and

$$\beta_{02} = ||sH(s)C(s)||_{\mathscr{L}_1}(||r||_{\mathscr{L}_\infty} + 2\Delta) + ||sH(s)(1 - C(s))||_{\mathscr{L}_1}(L\rho + L_0). \quad (3.24)$$

Since we have that $H(s)$ and $H_1(s)$ are BIBO stable and strictly proper, $||sH(s)(1 - C(s))||_{\mathscr{L}_1}$, $||sH_1(s)||_{\mathscr{L}_1}$, and $||sH(s)C(s)||_{\mathscr{L}_1}$ are finite.

Output Predictor: Consider the output predictor [15]:

$$\dot{\hat{y}}(t) = -m\hat{y} + m(u(t) + \hat{\sigma}(t)), \hat{y}(0) = 0. \quad (3.25)$$

Adaptive Law: The adaptive estimate $\hat{\sigma}(t)$ is given by

$$\dot{\hat{\sigma}}(t) = \Gamma\text{Proj}\left(\hat{\sigma}(t), -mP\tilde{y}(t)\right), \ \tilde{y} = \hat{y} - y, \ \hat{\sigma}(0) = 0, \quad (3.26)$$

where $P > 0$ is arbitrary and Γ is the adaptation rate and is subject to the following lower bound:

$$\Gamma > \max\{\frac{\alpha\beta_3^2}{(\alpha - 1)^2\beta_4 P}, \frac{\alpha\beta_4}{P\bar{\gamma}^2}\} \quad (3.27)$$

where $\alpha > 1$ is an arbitrary constant. The projection bound is

$$|\hat{\sigma}(t)| \leq \Delta \quad (3.28)$$

Control Law: The control law that generates the input signal is given as:

$$u(s) = C(s)(r(s) - \hat{\sigma}(s)). \quad (3.29)$$

Performance: In the following theorem, excerpted from [15], we summarize the guaranteed performance of the system when controlled by the $\mathscr{L}_1$-adaptive controller.

Theorem 3. *Given the system in* (2.4) *and the $\mathscr{L}_1$-adaptive output feedback controller given by* (3.25)*,* (3.26)*, and* (3.29) *subject to* (3.11)*, we have*

$$||\tilde{y}||_{\mathscr{L}_\infty} < \gamma_0 \quad (3.30)$$

$$||y - y_{ref}||_{\mathscr{L}_\infty} \leq \gamma_1 \quad (3.31)$$

16

$$||u - u_{ref}||_{\mathscr{L}_\infty} \leq \gamma_2 \tag{3.32}$$

where $\tilde{y}(t) = \hat{y}(t) - y(t)$, $\gamma_0 = \sqrt{\alpha\beta_4/(\Gamma P)}$ and

$$\gamma_1 = ||C(s)H(s)/M(s)||_{\mathscr{L}_1}\gamma_0/(1 - ||G(s)||_{\mathscr{L}_1}L), \tag{3.33}$$

$$\gamma_2 = L||H(s)C(s)/M(s)||_{\mathscr{L}_1}\gamma_1 + ||H_2(s)/M(s)||_{\mathscr{L}_1}\gamma_0. \tag{3.34}$$

Again, Theorem 3 shows that as the adaptation rate Γ is increased, the bounds on the errors decrease, thus improving performance.

A block-diagram reprsentation of the implementation of this control scheme is given in Figure 3.2
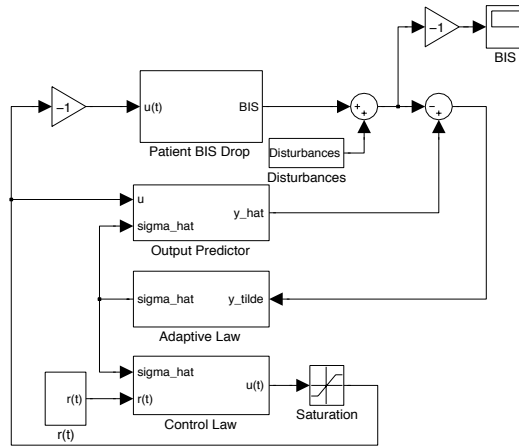
Figure 3.2: Output feedback controller Simulink model

CHAPTER 4

MODELING AND SYSTEM IDENTIFICATION

As this is an initial investigation of the applicability of $\mathscr{L}_1$-adaptive techniques, we have used clinical trial data from our earlier studies ([12, 13, 33]) in order to facilitate the comparison of controller performance results; ongoing efforts include design and analysis using surgical data records. The original trial was designed to define the relation between clinical evaluation of the state of conciousness, explicit recall, drug concentrations and BIS effects of the anesthetic agent isoflurane when administered alone to healthy volunteers under controlled conditions. Additionally, a series of external stimuli, or disturbances, were applied to the patients (volunteers) throughout the administration of anesthesia. These stimuli included: laryngeal mask insertion and removal; evoked potential evaluations involving the application of short electrical stimulation signals to the wrist of the volunteer at a period of every 3 seconds and up to 100 μA and 100 V amplitude; and *alertness evaluations* which included yelling at, shaking, and squeezing the trapezius muscle of the volunteer. Time-synchronized output measurements of the patients' BIS, mean arterial pressure (MAP) and HR were recorded every two seconds. For healthy individuals, normal ranges for MAP are between 70 and 110 mmHg, and the average resting HR for normal adults is around 70 beats per minute [34]. We previously developed quantitative models of the stimuli applied to the patients during the study [12] for use in system identification. Similar quantitative models have been adopted by other researchers [35], [36] and [20].

An example of a set of data taken from one subject during the clinical trial is shown in Figures 4.1 and 4.2. This data is fairly representative of the response expected from healthy volunteers to anesthesia and stimuli, however, as to be expected individual responses exhibit noticeable variation. This inter-patient variability is one of the main motivations for considering adaptive control techniques.

In previous work we found that switched-linear models effectively capture the response to anesthesia (note the effective switching that appears in the BIS levels in the plot presented in Figure 4.2). The constituent subsystems in these earlier

18
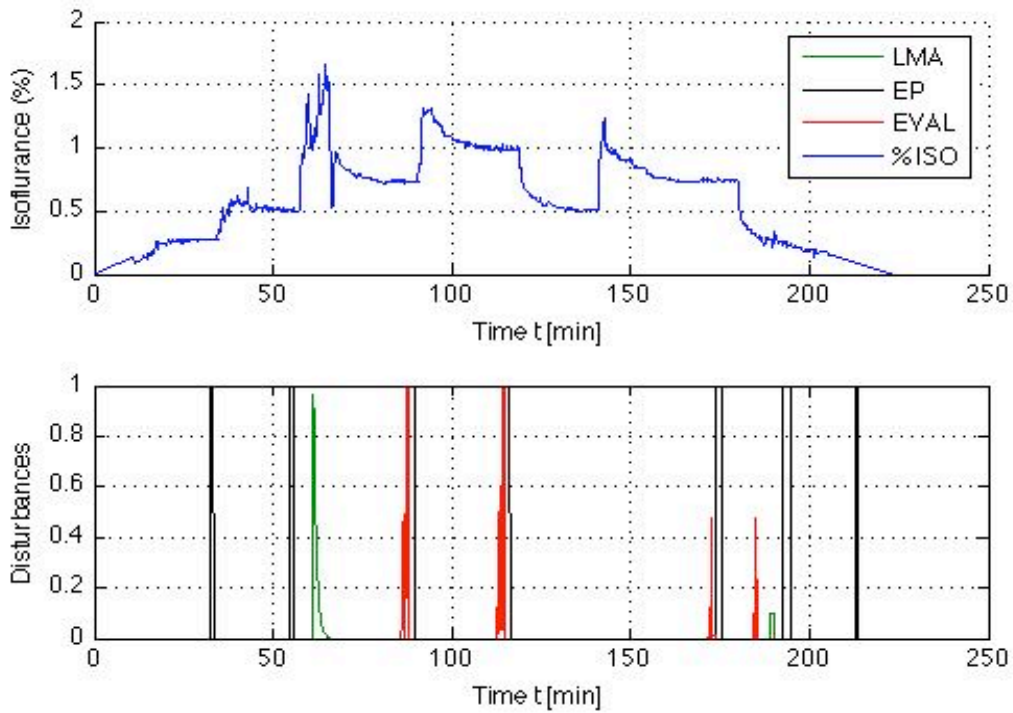
Figure 4.1: Isoflurane and Stimuli Inputs versus Time

switched-linear models were constructed using subspace identification methods applied to the data described above (specifically, we implemented the N4SID algorithm, first introduced in [37], in MATLAB; this provides *black box* models). Observed BIS values were used to choose between one of two models for a patient's response to anesthesia and stimuli (i.e., alert models, and sedated models that include both moderate and deeply sedated states). Switching between these two models was based on a BIS threshold value of 70, which was physiologically motivated (i.e., in [2] it is demonstrated that approximately 50% of the population will be unconscious at a BIS value of 70). For comparison purposes, we also constructed models of the dosing and related effects of isoflurane using the standard pharmacological approaches, i.e., PK-PD models; for details on these modeling results see [12].

19
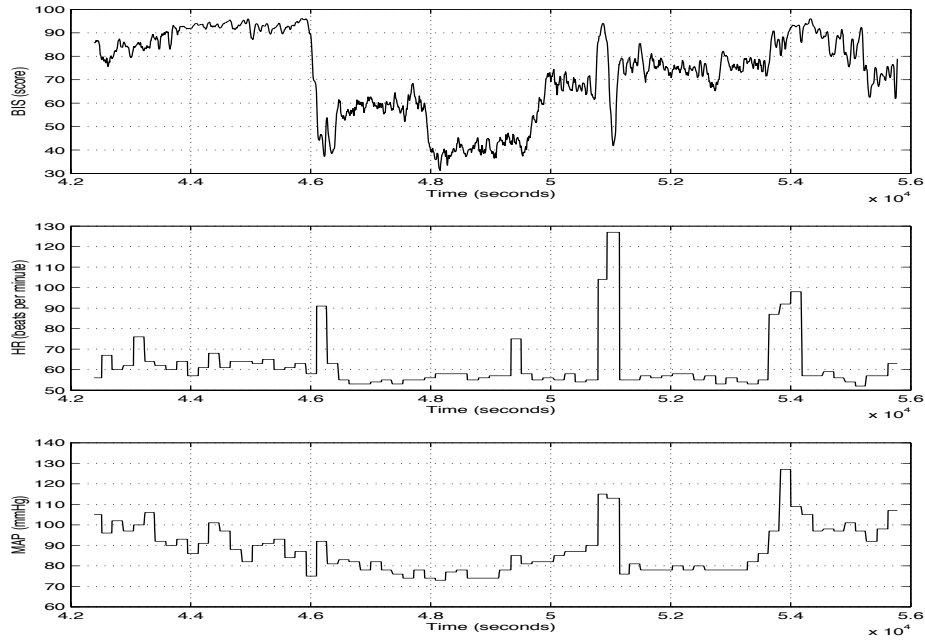
Figure 4.2: BIS, HR and MAP Outputs versus Time

4.1 Identification for $\mathscr{L}_1$-Adaptive Control State Feedback Models

In the work described herein, we construct *grey box* state-space models of patient response to anesthesia and stimuli in the sedated and alert states. Grey box methods are used to ensure that the input and output matrices (i.e., the *B* and *C* matrices) match, conforming with the $\mathscr{L}_1$-adaptive state feedback method requirements. To compute these models, the MATLAB *idgrey* command is used, ensuring the patient models have the required structure for the $\mathscr{L}_1$-adaptive control architecture. To date, models have been identified for patient datasets referred to as Patients 1, 2, 3, 5, 6 and 7. Estimation and validation results for these datasets are acceptable, with an overall average normalized residual error of approximately 34.5%. Examples are shown in Figures 4.3 and 4.4.

In the initial stage of the current study of identification and adaptive control design, we have constructed models and adaptive controllers for SISO systems (isoflurane input to BIS output), for which we evaluate and compare the modeling and controller performance results. We specifically focus on evaluating inter-

20
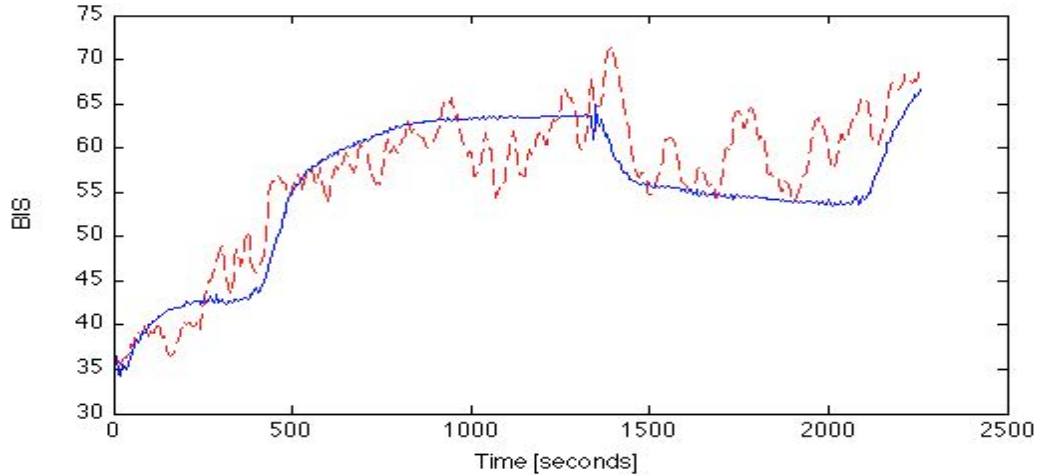
Figure 4.3: Patient 5 model validation results

patient adaptability of the controllers, and along these lines include performance analyses of applying controllers designed based on the model for one patient to other patient models. The second stage of this study is identification and adaptive control design evaluation for multi-input single-output (MISO) systems, namely, where external stimuli are included as disturbance inputs, in addition to the controlled isoflurane input.

The final stage of this project involves the construction of MIMO models and the application of multivariable $\mathscr{L}_1$-adaptive techniques to these models [38]. This stage will involve evaluating vital sign responses (HR and MAP, for example) as well as BIS responses. We note here that our focus is on automated control of patients primarily in the sedated state. Our assumption is that the attending physician performs the initial induction from alert to the lightly-sedated state in order to closely monitor initial patient response. Upon being lightly sedated and observed for safety reasons, the patient is then switched to the proposed automated control regime.

4.2   Identification of $\mathscr{L}_1$-Adaptive Control-Output Feedback Models

Unlike the state feedback $\mathscr{L}_1$-adaptive control architecture, *black box* models are sufficient for control design. Since this architecture relies on the transfer function,
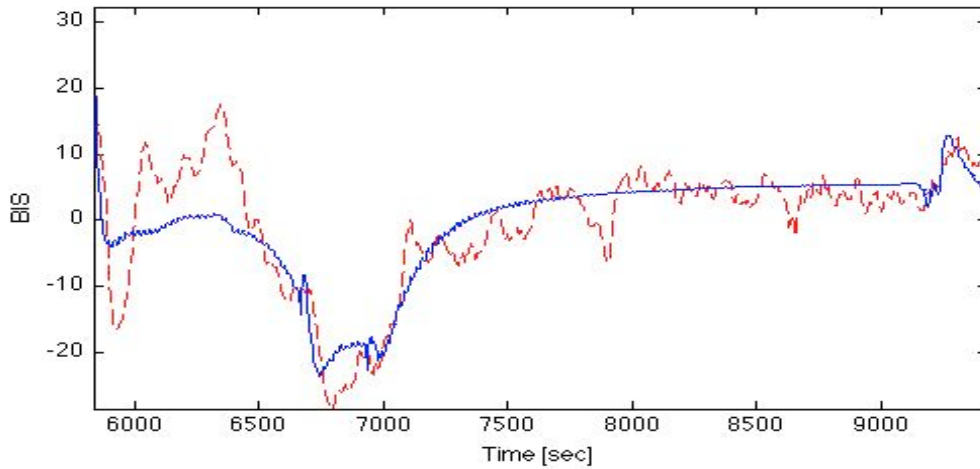
Figure 4.4: Patient 7 model performance

Table 4.1: Model performance

Patient	r_N
1	0.1843
2	0.3642
3	0.6058
5	0.334
6	0.3208
7	0.2681

the specific structure of the A, B, and C matrices are not required to be known. To acquire models for control design, the MATLAB command *n4sid* is used to calculate the models for 6 patients which we have designated Patients 1, 2, 3, 5, 6, and 7. The MATLAB code to compute these models can be seen in Appendix A.

CHAPTER 5

SIMULATION RESULTS: ANALYSIS AND DISCUSSION

In this chapter we present the design approaches for the state feedback and output feedback $\mathscr{L}_1$-adaptive control schemes and simulations results applying them to a set of patients named Patients 1, 2, 3, 5, 6, and 7. Additionally, performance differences between first and second order filters are investigated regarding their performance differences in the output feedback schemes. We conclude by discussing the differences between the $\mathscr{L}_1$-adaptive control approach and previous investigations using an LPV control scheme.

As noted previously, models were constructed through the MATLAB *idgrey* command on partitioned patient data (i.e., the data was divided roughly in half for estimation and validation purposes). A MATLAB m-file was created to identify the values for all parameters required for the system in (2.2), and for the $\mathscr{L}_1$-adaptive controller given by (3.1), (3.2), and (3.3). Simulink was then used to simulate the closed-loop systems. We have assumed there is no time delay in the system in our simulations to date (although the $\mathscr{L}_1$-adaptive control technique allows for a time delay margin bounded away from zero [32]).

5.1 State-Feedback Design Approach

The overall approach to designing the $\mathscr{L}_1$-adaptive state-feedback controller consisted of essentially two parts: choosing the matrix A_m and the filter $C(s)$.

The matrix A_m reflects our desired reference system behavior. That is, how we would like the system to behave. We design this matrix based on the previously discussed matching assumption, $A_m - A = B\theta^T$. Essentially, A_m can be designed using a variety of methods such as pole-placement or LQR-design. Our goal was that the system have minimal overshoot and a settling time between one and two minutes.

After A_m is chosen, we can then design $C(s)$. First, we select a conservative

23

convex set Θ that our estimates, $\hat{\theta}(t)$, reside in. From this, we calculate θ_{max} which is used to establish the upper bound on the $\mathscr{L}_1$-norm of $G(s)$. A filter $C(s)$ then can be chosen that is asymptotically stable, strictly proper, and has a DC gain $C(0) = 1$. For the following simulations, first order filters were used.

Having chosen A_m and $C(s)$, all that remains is to set the projection bounds and the adaptation rate Γ. The projection bounds are set based on the set Θ. The adaptation rate is normally set as high as the physical system will allow, as this improves tracking performance. For the majority of our simulations, Γ was set to 1000.

5.1.1   Patient 1 $\mathscr{L}_1$-Adaptive Control-State Feedback

The model parameters for *Patient 1-sedated state* are:

$$A_m = \begin{bmatrix} -0.0063 & -0.0413 & -0.0324 & -0.0334 \\ 0.0326 & -0.0775 & -0.1507 & -0.2732 \\ -0.0004 & -0.0081 & 0.1035 & 0.2851 \\ 0.2276 & 0.1392 & -0.4974 & -0.7736 \end{bmatrix},$$

$$B = \begin{bmatrix} -0.0047 \\ 0.0258 \\ 0.0004 \\ 0.4817 \end{bmatrix},$$

$$C = \begin{bmatrix} 136.01 \\ -9.7696 \\ 2.3834 \\ -0.9740 \end{bmatrix}, \text{ and } \theta = \begin{bmatrix} -0.4626 \\ -0.2693 \\ -0.1000 \\ 1.2022 \end{bmatrix}. \tag{5.1}$$

We assume the unknown vector θ belongs to a known compact set, which we have chosen initially to be $\Theta = \left(\theta \in \mathbb{R}^4 | \theta_i \in [-8, 8] \right)$. Typically, a compact set is chosen for Θ that is conservative with respect to the values these parameters are expected to take. The set is assumed compact for asymptotic stability considerations. The conservativeness of our choice for θ can be seen by the above values of θ required for the desired performance. We set $Q = 2\mathbb{I}$ to simplify calculations and $\Gamma = 1000$ to have a relatively high adaptation rate. The $\mathscr{L}_1$-adaptive controller given by (3.1), (3.2), and (3.3) is used to track a given reference $r(t)$. For the set Θ,

24

an appropriate value for L is 32, based on the $\mathscr{L}_1$-stability condition given in (3.4). That is, to satisfy the $\mathscr{L}_1$-gain stability condition, $C(s)$ must be chosen such that $\|\bar{G}(s)\|_{\mathscr{L}_1} < 1/32$. If we choose $C(s) = \frac{40}{s+40}$, then $\|\bar{G}(s)\|_{\mathscr{L}_1} = 0.0276 < 0.0312$, satisfying the necessary condition for stability. $C(s)$ is chosen in this manner since it has the relatively small bandwidth compared to the other first order filters that would satisfy the $\mathscr{L}_1$-stability condition. This is done to compensate should higher frequency signals enter $\bar{r}(t)$. Simulation results for the $\mathscr{L}_1$-adaptive controller applied to the dynamic response model for Patient 1 can be seen in Figure 5.1.

Figure 5.1: Patient 1 BIS reference tracking and control effort

5.1.2 Patient 2 $\mathscr{L}_1$-Adaptive Control-State Feedback

The model parameters for *Patient 2-sedated state* are:

$$
A_m = \begin{bmatrix} 0.0141 & -0.0364 & -0.0875 & -0.0528 \\ 0.0731 & -0.1146 & -0.2768 & -0.3042 \\ 0.0771 & -0.0500 & -0.3899 & 0.0408 \\ 0.0812 & -0.0837 & -0.5631 & -0.2635 \end{bmatrix},
$$

25

$$B = \begin{bmatrix} 0.0003 \\ 0.0010 \\ 0.0015 \\ 0.0015 \end{bmatrix},$$

$$C = \begin{bmatrix} 237.2070 \\ -11.6216 \\ 0.5634 \\ -2.9660 \end{bmatrix}, \text{ and } \theta = \begin{bmatrix} -53.5172 \\ -27.2409 \\ 272.6358 \\ 91.1382 \end{bmatrix}. \tag{5.2}$$

For Patient 2, it is assumed the unknown vector θ belongs to the compact set $\Theta = \left(\theta \in \mathbb{R}^4 | \theta_i \in [-275, 275] \right)$. Similar to the Patient 1 simulation, $Q = 2\mathbb{I}$ to simplify calculations, but, for Patient 2, $\Gamma = 100000$ is chosen as 1000 does not yield the performance we desire. The $\mathscr{L}_1$-adaptive controller given by (3.1), (3.2), and (3.3) is used to track a given reference $r(t)$. For our compact set Θ, $L = 1100$ based on the condition given in the $\mathscr{L}_1$-stability condition (3.4). To satisfy the $\mathscr{L}_1$-gain stability condition, $C(s)$ must be chosen such that $\|\bar{G}(s)\|_{\mathscr{L}_1} < 1/1100$. If we choose $C(s) = \frac{15}{s+15}$, $\|\bar{G}(s)\|_{\mathscr{L}_1} = 2.22 * 10^{-4} < 9.09 * 10^{-4}$ so the condition is satisfied. $C(s)$ is chosen in this manner to satisfy the $\mathscr{L}_1$-stability condition and to suppress high frequency that may be in the signal $\bar{r}(t)$. The simulation results of the $\mathscr{L}_1$ adaptive controller applied to the model for Patient 2 can be seen in Figure 5.2.

Figure 5.2: Patient 2 BIS reference tracking and control effort

5.1.3 Patient 3 $\mathscr{L}_1$-Adaptive Control-State Feedback

The model parameters for *Patient 3-sedated state* are:

$$A_m = \begin{bmatrix} -0.0197 & 0.0464 & -0.0425 & -0.1257 \\ -0.0335 & 0.1176 & -0.0390 & -0.2506 \\ -0.0362 & 0.1943 & -0.2665 & -0.6133 \\ -0.0870 & 0.3325 & -0.0259 & -0.5964 \end{bmatrix},$$

$$B = \begin{bmatrix} -0.0082 \\ -0.0185 \\ -0.0166 \\ -0.0381 \end{bmatrix},$$

$$C = \begin{bmatrix} 173.0765 \\ -9.0676 \\ 2.5768 \\ -2.4800 \end{bmatrix} \text{ and } \theta = \begin{bmatrix} -2.2918 \\ 9.0462 \\ -8.0825 \\ -16.3102 \end{bmatrix}. \tag{5.3}$$

For Patient 3, it is assumed the unknown vector θ belongs to the compact set

27

$\Theta = (\theta \in \mathbb{R}^4 | \theta_i \in [-20, 20])$. For ease of calculation, we set $Q = 2\mathbb{I}$ and set $\Gamma = 1000$ to have a reasonably high adaptive gain. The $\mathscr{L}_1$-adaptive controller is deisgned based on (3.1), (3.2), and (3.3). For our compact set Θ, L is set to 80 as defined in the $\mathscr{L}_1$-stability condition (3.4). We then must choose $C(s)$ to satisfy the $\mathscr{L}_1$-gain stability condition. That is, $C(s)$ must be chosen such that $\|\bar{G}(s)\|_{\mathscr{L}_1} < 1/80$. If we choose $C(s) = \frac{15}{s+15}$, $\|\bar{G}(s)\|_{\mathscr{L}_1} = 4.52 * 10^{-3} < 1/80$ so the condition is satisfied. This $C(s)$ filter is chosen such that it suppresses high frequencies that would otherwise enter the system, and it is also used for other controllers. This is useful for comparison purposes and analysis purposes when evaluating performance of among the patients and controller combinations. The simulation results of the $\mathscr{L}_1$-adaptive controller applied to the model for Patient 3 can be seen in Figure 5.3.

Figure 5.3: Patient 3 BIS reference tracking and control effort

28

5.1.4 Patient 5 $\mathscr{L}_1$-Adaptive Control-State Feedback

The model parameters for *Patient 5-sedated state* are:

$$A_m = \begin{bmatrix} -0.0553 & 0.0037 & 0.0329 & -0.0265 \\ 0.6119 & -0.5097 & 0.1335 & 0.2688 \\ 0.9460 & -0.6811 & -0.0161 & 0.5125 \\ -0.6282 & 0.3854 & -0.1738 & -0.1729 \end{bmatrix},$$

$$B = \begin{bmatrix} -0.0070 \\ 0.0833 \\ 0.1311 \\ -0.0873 \end{bmatrix},$$

$$C = \begin{bmatrix} 139.8580 \\ -9.8886 \\ 0.4684 \\ -0.5230 \end{bmatrix}, \text{ and } \theta = \begin{bmatrix} 7.2593 \\ -4.8310 \\ -0.0168 \\ 1.7065 \end{bmatrix}. \tag{5.4}$$

For this patient model, we have chosen the compact set for θ initially to be $\Theta = \left(\theta \in \mathbb{R}^4 | \theta_i \in [-15, 15] \right)$. The conservativeness of our choice for θ can be seen by the above values of θ required for the desired performance. We set $Q = 2\mathbb{I}$ and $\Gamma = 1000$ just as was done for Patients 1 and 3. The $\mathscr{L}_1$-adaptive controller is designed based on (3.1), (3.2), and (3.3). For the set Θ, an appropriate value for L is 60, based on the $\mathscr{L}_1$-stability condition (3.4), that is, to satisfy the $\mathscr{L}_1$-gain stability condition, $C(s)$ must be chosen such that $\|\bar{G}(s)\|_{\mathscr{L}_1} < 1/60$. For the filter $C(s)$, we choose $C(s) = \frac{15}{s+15}$. Then $\|\bar{G}(s)\|_{\mathscr{L}_1} = 0.0162 < 0.0167$, satisfying the necessary condition for stability, and this filter helps reject high frequencies that may appear in the signal $\bar{r}(t)$. Simulation results for the $\mathscr{L}_1$-adaptive controller applied to the dynamic response model for Patient 5 can be seen in Figure 5.4.
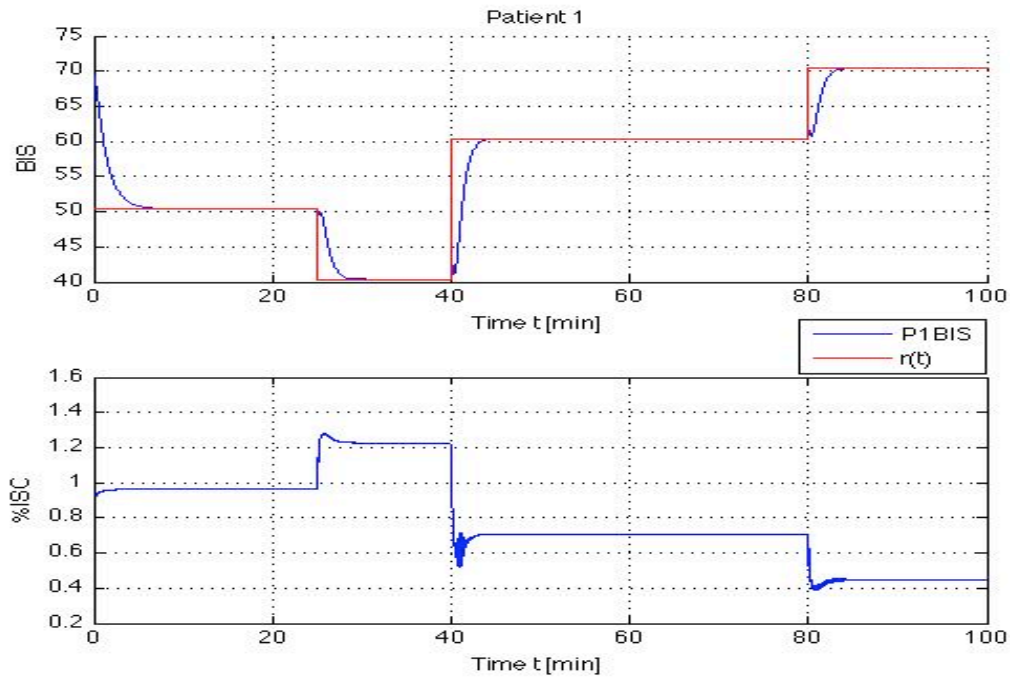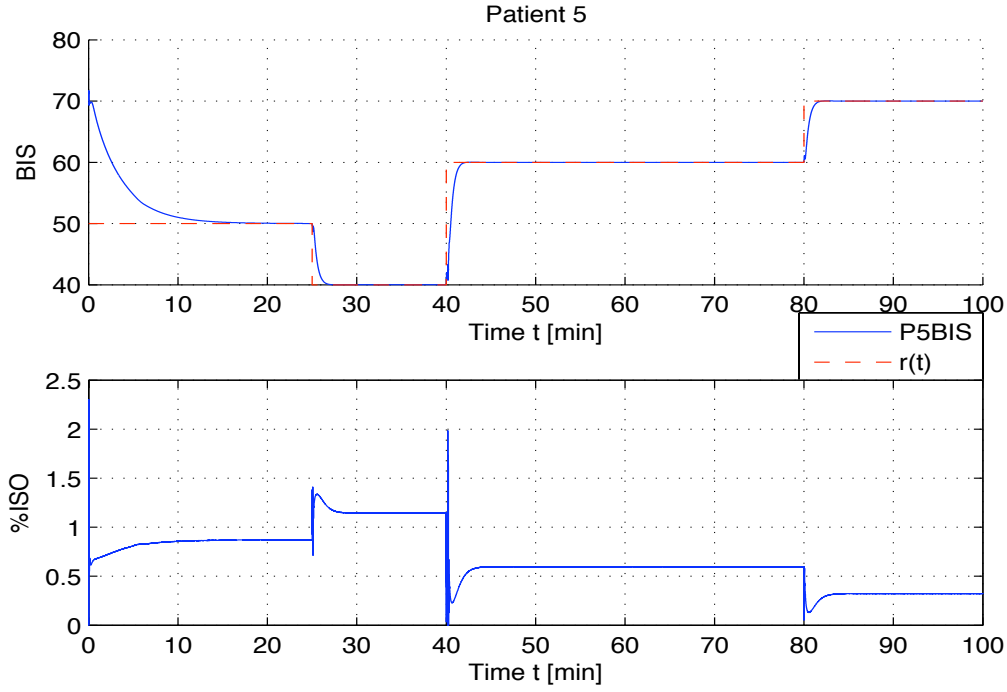
Figure 5.4: Patient 5 BIS reference tracking and control effort

5.1.5 Patient 6 $\mathscr{L}_1$-Adaptive Control-State Feedback

The model parameters for *Patient 6 – sedated state* are:

$$A_m = \begin{bmatrix} 0.0309 & 0.0058 & -0.1055 & -0.0181 \\ 0.1087 & -0.0247 & -0.1641 & 0.0147 \\ 0.3999 & -0.0985 & -0.3796 & -0.5498 \\ 0.2189 & -0.1152 & -0.0106 & -0.3807 \end{bmatrix},$$

$$B = \begin{bmatrix} 0.0025 \\ 0.0072 \\ 0.0267 \\ 0.0151 \end{bmatrix},$$

$$C = \begin{bmatrix} 137.6774 \\ -8.9751 \\ -5.8180 \\ 3.9340 \end{bmatrix}, \theta = \begin{bmatrix} -14.5340 \\ 1.5157 \\ 7.6818 \\ 21.4511 \end{bmatrix} \tag{5.5}$$

For Patient 6, the unknown vector θ is assumed to belong to the compact set

30

$\Theta = \left(\theta \in \mathbb{R}^4 | \theta_i \in [-25, 25] \right)$. Let $Q = 2\mathbb{I}$ and $\Gamma = 1000$. The $\mathscr{L}_1$-adaptive controller given by (3.1), (3.2), and (3.3) is used to track a given reference $r(t)$. For our compact set Θ, $L = 100$ based on the condition given in the $\mathscr{L}_1$-stability condition (3.4). To satisfy the $\mathscr{L}_1$-gain stability condition, $C(s)$ must be chosen such that $\|\bar{G}(s)\|_{\mathscr{L}_1} < 0.01$. If we choose $C(s) = \frac{15}{s+15}$, then this filter gives us $\|\bar{G}(s)\|_{\mathscr{L}_1} = 3.4 * 10^{-3} < 0.01$ satisfying the stability condition. It also suppresses high frequencies in $\bar{r}(t)$ and allows us to compare patient and controller combinations. The simulation results of the $\mathscr{L}_1$-adaptive controller applied to the model for Patient 6 can be seen in Figure 5.5.
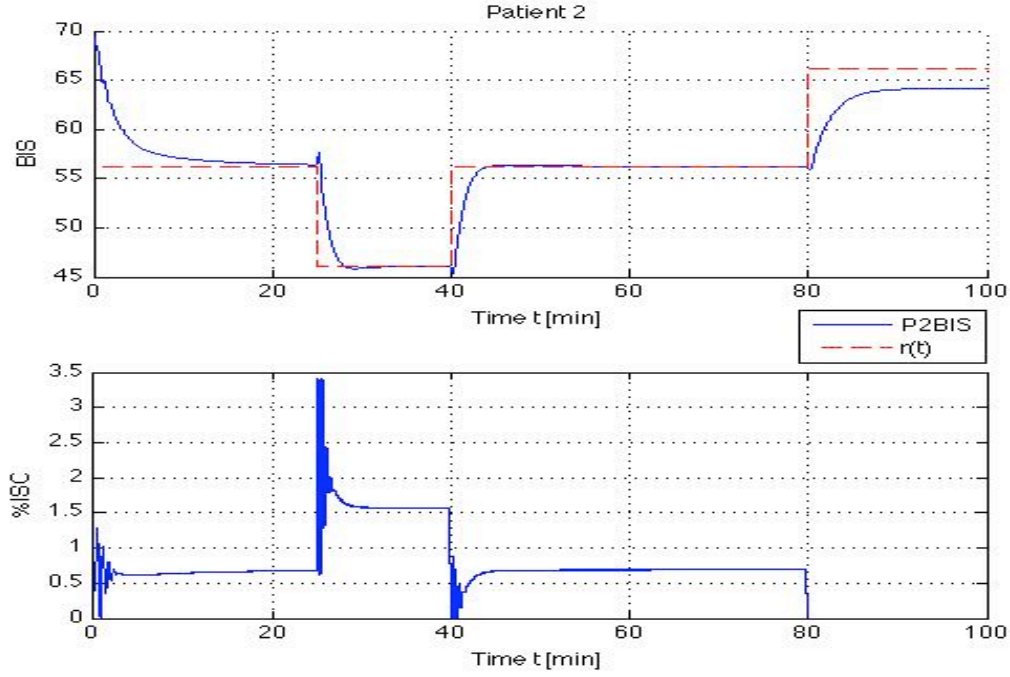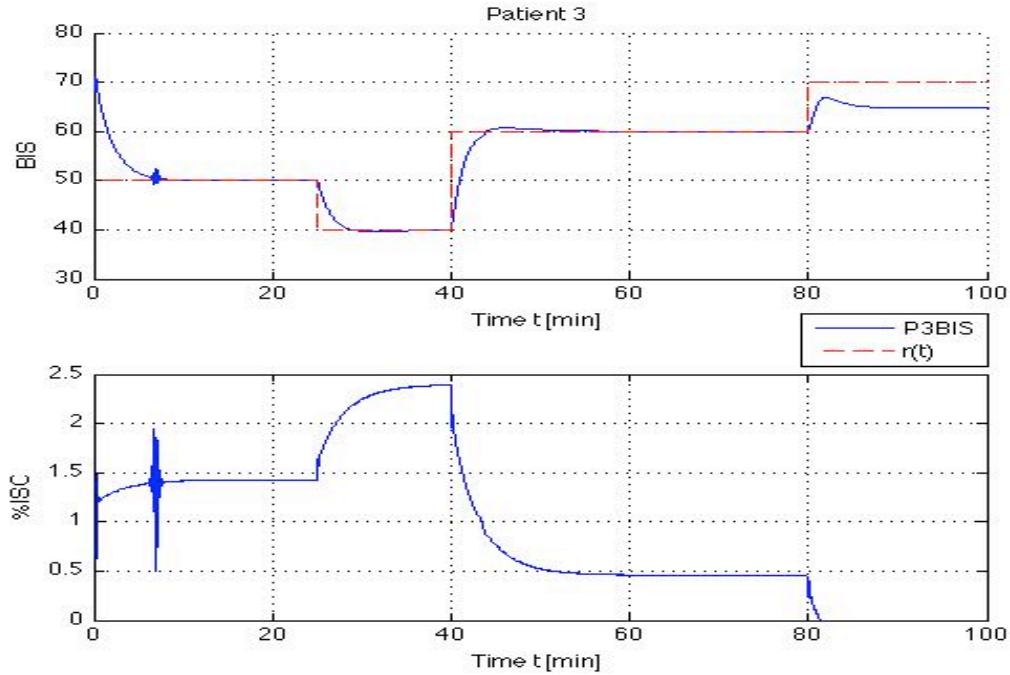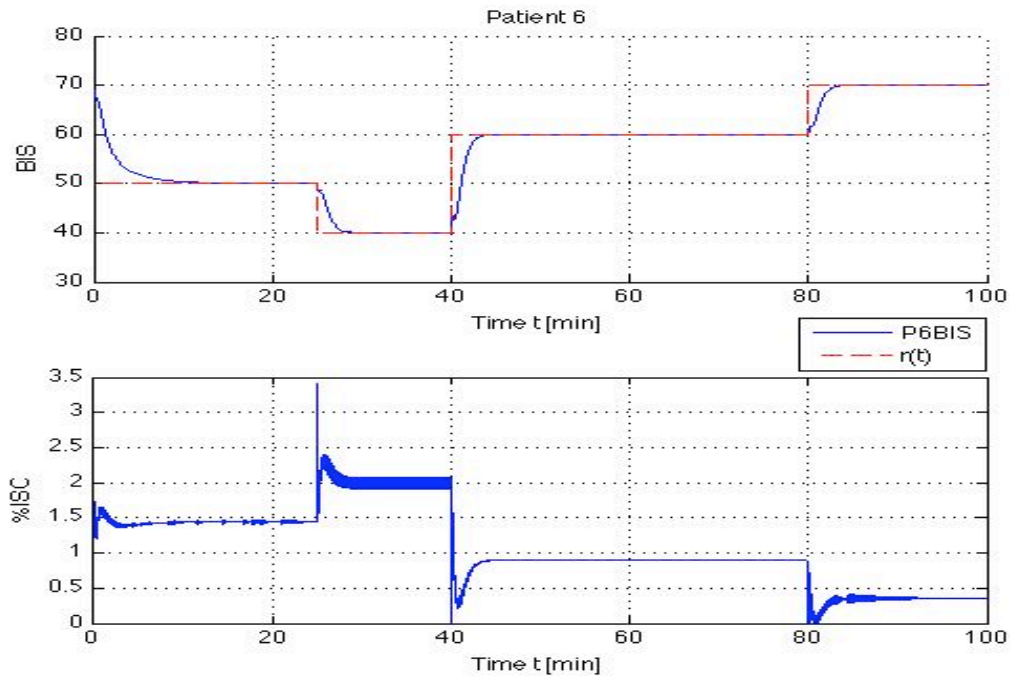

Figure 5.5: Patient 6 BIS reference tracking and control effort

5.1.6 Patient 7 $\mathscr{L}_1$-Adaptive Control-State Feedback

The model parameters for *Patient 7 – sedated state* are:

$$A_m = \begin{bmatrix} -0.0532 & 0.0324 & -0.0979 & 0.0392 \\ -0.232 & 0.232 & -0.5170 & 0.1317 \\ -0.3785 & 0.4849 & -0.6837 & 0.1086 \\ 0.2636 & -0.2993 & 0.7114 & -0.2941 \end{bmatrix},$$

31

$$B = \begin{bmatrix} -0.0016 \\ -0.0078 \\ -0.0118 \\ 0.0082 \end{bmatrix},$$

$$C = \begin{bmatrix} 175.3789 \\ -10.2101 \\ -0.8136 \\ 0.3883 \end{bmatrix}, \theta = \begin{bmatrix} 32.0021 \\ -39.6487 \\ 56.6002 \\ -32.3687 \end{bmatrix} \tag{5.6}$$

For Patient 7, it is assumed the unknown vector θ belongs to the compact set $\Theta = \left(\theta \in \mathbb{R}^4 | \theta_i \in [-75, 75] \right)$. Similar to our previous Patient 2 simulation, we set $Q = 2\mathbb{I}$ and $\Gamma = 10000$. The $\mathscr{L}_1$-adaptive controller given by (3.1), (3.2), and (3.3) is implemented to track a given reference BIS trajectory, $r(t)$. For our compact set Θ, $L = 300$ based on the condition given in the $\mathscr{L}_1$-stability condition (3.4). Again, we must choose $C(s)$ such that it satisfies the $\mathscr{L}_1$-gain stability condition. That is, $C(s)$ must be chosen such that $\|\bar{G}(s)\|_{\mathscr{L}_1} < 1/300$. If we choose $C(s) = \frac{15}{s+15}$, $\|\bar{G}(s)\|_{\mathscr{L}_1} = 1.4 * 10^{-3} < 1/300$ so the condition is satisfied as required and the system can suppress high frequencies in the $\bar{r}(t)$ signal. The simulation results of the $\mathscr{L}_1$-adaptive controller applied to the model for Patient 7 can be seen in Figure 5.6.

As can be seen in Figures 5.1-5.6, the $\mathscr{L}_1$ control designs achieve the desired tracking performance quickly. The tracking errors are small in all examples. The metric used to evaluate BIS reference tracking performance is the normalized mean square error, or residual error term,

$$r_N = \frac{\sum_{n=1}^{N} (y_n - \hat{y}_n)^2}{\sum_{n=1}^{N} y_n^2} \tag{5.7}$$

where the reference signal y and the patient's BIS value $\hat{y}$ contain N data points over the course of the simulation. The residual errors, r_N, can be seen in Table 5.1; the average residual error value is approximately 0.0031.

In each case, the system has a non-zero initial condition which leads to an exponentially decaying term in the control and system state signal; this does not affect the performance of the system throughout. The same design approach was followed for Patients 1, 2, 3, 5, 6 and 7 and yielded similar results.

32
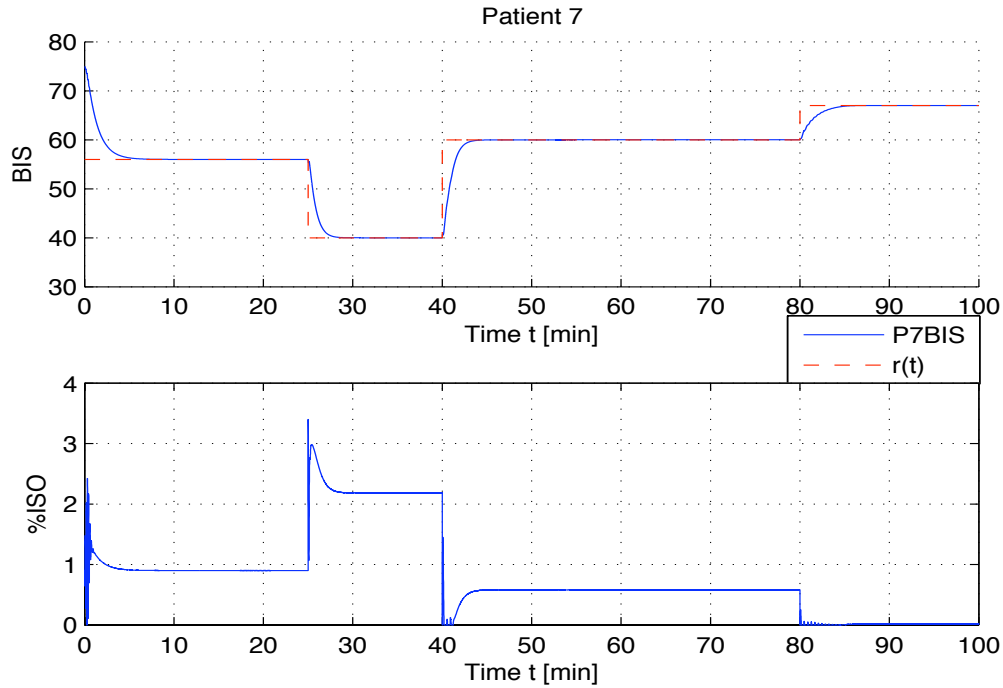
Figure 5.6: Patient 7 BIS reference tracking and control effort

5.1.7 State-Feedback Robustness to Inter-patient Variability

After control designs were completed for each of the Patients 1, 2, 3, 5, 6 and 7, they were then simulated in feedback with models for the remaining patients to evaluate inter-patient design robustness. We simulated the control designs for Patients 1, 2, 3, 5, 6, and 7 on each of the other patient models. That is, we applied the controller designed for Patient 1 on Patients 2, 3, 5, 6, and 7 and similarly for all other patient-controller combinations. In general, the patients' BIS levels tracked the desired reference profile closely, regardless of whether the applied controller was that specifically designed for them or not.

Remark 3. *Note that in earlier studies we found that the patient datasets could be separated into two subgroups based on similar dynamic response behavior; one group included Patients 1, 5, 6, and 7, and the second group included Patients 2 and 3 [12].*

Figures 5.7 and 5.8 show two simulations demonstrating that the $\mathcal{L}_1$ controller designs achieve performance that is close to, and in some cases better than, the performance achieved on the patient model used to design the controller. In these examples, the controller design based on the model for Patient 1 is applied to the

33

model for Patient 2 (one of the second subgroup patient datasets; see remark 3), and the controller designed based on the model for Patient 5 is applied to the model for Patient 1. As can be seen in the plots, both patient responses achieve nearly the same BIS profile as the patient for whom the controller was designed. Computing the residual errors, given by (5.7), indicates that the tracking profile for Patient 2 is close to that for Patient 1 using the controller designed for the Patient 1 model; that is, their respective r_N values are 0.0030 and 0.0026. Similarly, evaluating the tracking profile of Patient 1 with the controller designed for Patient 5 yields a residual error value of $r_N = 0.0029$ while for the Patient 5 model the residual error using the Patient 5 controller is 0.0032.
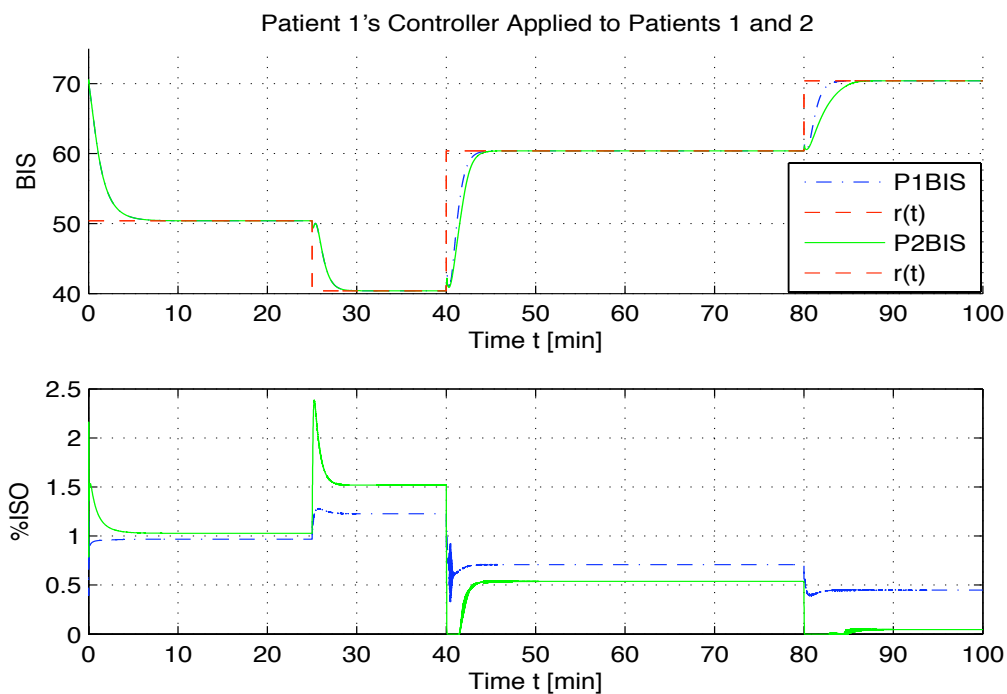
Figure 5.7: BIS reference tracking: Patient 1 controller applied to Patient 2 model

Based on using the normalized mean square tracking error (5.7) as our performance metric, the controller/patient combination that performs best for our desired reference profile is the Patient 1 controller applied to the Patient 7 model, resulting in $r_N = 0.002$. The worst controller/patient combination performance results from applying the Patient 7 controller to the model for Patient 5, giving $r_N = 0.0067$. The average r_N value over all combinations is 0.0033, which is very close to the average r_N value resulting from using the controllers designed specifically for patients, which is $r_N = 0.003$. These results demonstrate the inherent
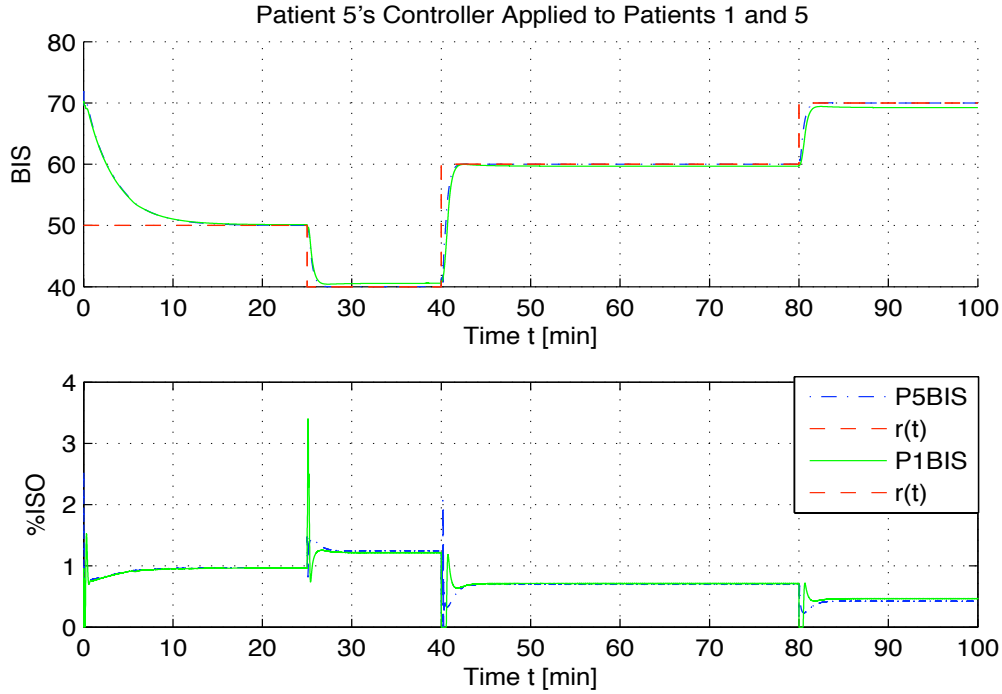
Figure 5.8: BIS reference tracking: Patient 5 controller applied to Patient 1 model

robustness properties of the $\mathscr{L}_1$-adaptive methods. Based on these r_N values, it also appears that the Patient 3 controller provides superior tracking performance over all control designs completed to date. This controller results in the lowest r_N value when applied to the models for Patients 5 and 6 (0.0026 and 0.0024, respectively) and the lowest average r_N value when applied to the other patients, 0.0026. Table 5.1 provides a summary of the tracking performance results.

Table 5.1: Normalized residual errors: BIS tracking performance

Patient/Controller	1	2	3	5	6	7
1	0.0026	N/A	0.0029	0.0029	0.0029	0.0028
2	0.0030	0.0021	0.0029	0.0043	0.0031	0.0037
3	N/A	N/A	0.0044	N/A	N/A	N/A
5	0.0032	N/A	0.0026	0.0032	0.0031	0.0067
6	0.0025	N/A	0.0024	0.0049	0.0029	0.0058
7	0.002	N/A	0.0023	0.0045	0.0031	0.0033

A second metric used to evaluate controller performance is the total amount of isoflurane required to attain the tracking performance. (Note that high total

35

amounts of anesthetics used in surgery have been associated with negative long-term patient outcomes [39, 40, 41]). Table 5.2 provides relative quantities, in liters, of isoflurane required to maintain the BIS tracking profiles previously discussed. Note that inhalational anesthetics are delivered as a percentage by volume to an external respiratory circuit. This gas mixture is then delivered to the lungs by a ventilation system, measured in liters/minute. So, for example if the total fresh gas flow mixture from the machine is $2 - 4$ l/min, and the isoflurane is 1% (at an assumed mean alveolar concentration (MAC) of 1), then $20 - 40$ cc/min of isoflurane gas is being delivered to the respiratory circuit. To estimate the following relative quantities of isoflurane used, we assumed an average fresh gas flow mixture delivery rate of 3 l/min at 1 MAC. The average isoflurane use over this patient set is 2.593 liters; average isoflurane use for a comparable BIS reference and simulation time resulting from previous linear parameter-varying control studies was approximately 3.211 liters [42].

Table 5.2: Control effort: Isoflurane use in liters

Patient/Controller	1	2	3	5	6	7
1	2.391	N/A	2.928	2.368	2.969	2.697
2	2.125	2.035	2.727	2.121	3.206	2.635
3	N/A	N/A	2.754	N/A	N/A	N/A
5	2.354	N/A	2.932	2.338	3.011	2.705
6	2.290	N/A	2.820	2.366	3.248	2.734
7	2.086	N/A	2.769	2.036	3.259	2.793

5.2 Output-Feedback Design Approach

The overall objective for designing these controllers was to have the patient BIS signal follow a desired BIS trajectory and be robust. To ensure stability of the closed-loop system when using controllers we designed, the design process started by ensuring that the $\mathscr{L}_1$-stability condition will be satisfied. To simplify the design process, the average BIS value from identified model is detrended to zero and the models are then considered as transfer functions of the magnitude of *BIS decrease* rather than exact BIS levels. Essentially the models values are shifted. This does not affect the stability of the system.

The desired reference transfer function, $M(s)$, is then selected based on the desired patient behavior using (2.7). Values of m that were investigated were 1/15, 1/30, and 1/60, as these would give settling times of around 1 minute, 2 minutes, and 4 minutes, respectively, when r(t) follows step like behavior. A settling time of one minute was desired and was achieved with many patients, but to ensure robustness, other values were examined.

In terms of designing the $\mathscr{L}_1$-adaptive output-feedback controllers for the patient set, the greatest difficulty lay in ensuring $H(s)$ is BIBO stable, since we have non-minimum phase zeros. We used first order filters $C(s)$ and first order desired reference models $M(s)$, then examined the denominator of $H(s)$ in terms of ω and m to reduce the possible choices for ω and m. In particular, we took advantage of the fact that with first order choices,

$$H(s) = \frac{C_d(s)M_n(s)A_n(s)}{M_d(s)C_n(s)A_n(s) + (C_d(s) - C_n(s))M_n(s)A_d(s)} \tag{5.8}$$

becomes the much more simple form given in the equation below, i.e.,

$$H(s) = \frac{(s + \omega)M_n(s)A_n(s)}{\omega M_d(s)A_n(s) + sM_n(s)A_d(s)}. \tag{5.9}$$

The filter $C(s)$ needs to be chosen in combination with $M(s)$ such that the $\mathscr{L}_1$-stability condition is satisfied. The parameters m and ω were selected using a combination of classical stability analysis methods and trial and error. MATLAB was used to verify stability of $H(s)$ and the code can be seen in Appendix C. Finally, the projection initial condition is set to zero and the adaptive rate, Γ, and projection bounds are set conservatively large.

5.2.1 Patient 1 $\mathscr{L}_1$-Adaptive Control-Output Feedback

As previously noted, models were constructed through the MATLAB *n4sid* command on partitioned patient data (i.e., the data was divided roughly in half for estimation and validation purposes). A MATLAB m-file was created to identify the values for all parameters required for the system in (2.4), and for the $\mathscr{L}_1$-adaptive controller given by (3.25), (3.26), and (3.29). Simulink was then used to simulate the closed-loop systems.

37

5.2.1.1 Tracking, no disturbances

The model parameters for *Patient 1-sedated state* are:

$$A = \begin{bmatrix} -0.0042 & -0.0401 & -0.0319 & -0.0391 \\ 0.0207 & -0.0844 & -0.1533 & -0.2422 \\ -0.0006 & -0.0082 & 0.1034 & 0.2855 \\ 0.0048 & 0.0095 & -0.5456 & -0.1946 \end{bmatrix},$$

$$B = \begin{bmatrix} -0.0047 \\ 0.0258 \\ 0.0004 \\ 0.4817 \end{bmatrix},$$

$$C = \begin{bmatrix} 136.01 \\ -9.7696 \\ 2.3834 \\ -0.9740 \end{bmatrix}. \tag{5.10}$$

The transfer function for the ISO/BIS model of *Patient 1-sedated state* is:

$$\frac{\text{BIS}}{\text{ISO}} : \frac{-1.359s^3 - 1.362s^2 + 0.08081s - 0.00635}{s^4 + 0.1798s^3 + 0.1461s^2 + 0.01316s + 0.0001649} \tag{5.11}$$

We have assumed there is no time delay in the system in our simulations. The transfer function for ISO to BIS is assumed to be strictly proper (the models identified using *n4sid* are just that). We set $P = 1$, $\Delta = 100$ and $\Gamma = 50000$. The $\mathscr{L}_1$-adaptive controller given by (3.25), (3.26), and (3.29) is used to track a given reference BIS trajectory, $r(t)$. If, using (2.7), we set $m = 1/30$, and we set $\omega = 0.001$ in (3.9), we can show that (3.10) is strictly proper and BIBO stable as required. Simulation results for the $\mathscr{L}_1$-adaptive controller applied to the dynamic response model for Patient 1 can be seen in Figure 5.9.

As can be seen in Figure 5.9, the $\mathscr{L}_1$-adaptive output feedback control design achieves the desired tracking performance quickly, and the tracking error is small. The metric used to evaluate BIS reference tracking performance is the normalized mean square error, or residual error term,

$$r_N = \frac{\sum_{n=1}^{N}(y_n - \hat{y}_n)^2}{\sum_{n=1}^{N} y_n^2} \tag{5.12}$$

38
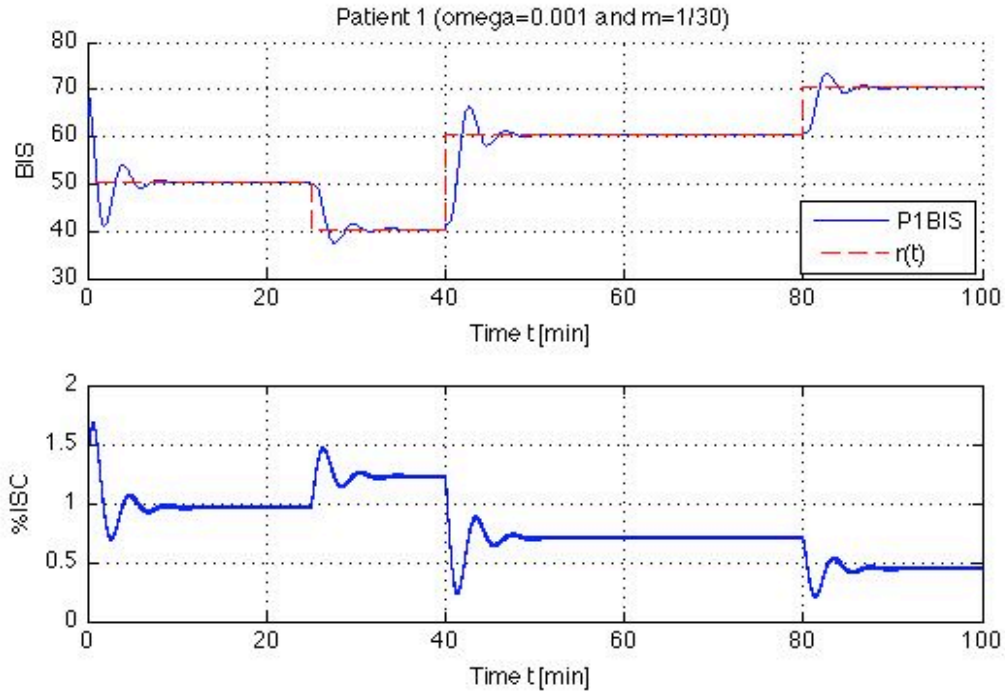
Figure 5.9: Patient 1 output feedback controller with $\Gamma = 50000$

where the reference signal y and the patient's BIS value $\hat{y}$ contain N data points over the course of the simulation. The residual error, r_N for Patient 1 with the previously discussed parameters is 0.0019. Other design results can be seen in Tables 5.3 and 5.4.

Table 5.3: Patient 1 normalized residual errors: BIS tracking performance

ω/m	1/15	1/30	1/60
0.002	N/A	0.0027	0.0047
0.001	0.0042	0.0019	0.0022
0.0005	0.0039	0.0035	0.0027
0.0001	0.0067	0.0070	0.0069

A second metric used to evaluate controller performance is the total amount of isoflurane required to attain the tracking performance. As noted before, larger amounts of anesthetics used in surgery have been associated with negative long-term patient outcomes [39, 40, 41]. Inhalational anesthetics, like isoflurane, are delivered as a percentage by volume to an external respiratory circuit. This gas mixture is then delivered to the lungs by a ventilation system, measured in liters/minute.

39

Table 5.4: Patient 1 control effort: Isoflurane use in liters

ω/m	1/15	1/30	1/60
0.002	N/A	2.401	2.408
0.001	2.401	2.404	2.410
0.0005	2.406	2.409	2.416
0.0001	2.453	2.456	2.463

So, for example if the total fresh gas flow mixture from the machine is $2-4$ l/min, and the isoflurane is 1% (at an assumed mean alveolar concentration (MAC) of 1), then $20-40$ cc/min of isoflurane gas is being delivered to the respiratory circuit. To estimate the relative quantities of isoflurane used, we assumed an average fresh gas flow mixture delivery rate of 3 l/min at 1 MAC. The approximate isoflurane use for Patient 1 is 2.40403 liters; isoflurane use for a comparable BIS reference and simulation time resulting from previous linear parameter-varying control studies [42] was approximately 2.745 liters. The smaller amount of isoflurane used by the $\mathscr{L}_1$-adaptive output feedback controller would be better for the patient than the higher amount used by the LPV controller designed for Patient 1.

5.2.1.2 Tracking, with disturbances

We will treat the EP/BIS, EVAL/BIS, and LMA/BIS as disturbances to our ISO/BIS system. The transfer functions for the disturbances model of *Patient 1-sedated state* are:

$$\frac{\text{BIS}}{\text{EP}} : \frac{-0.2798s^3 - 0.3321s^2 - 0.01559s - 0.0008389}{s^4 + 0.1798s^3 + 0.1461s^2 + 0.01316s + 0.0001649} \tag{5.13}$$

$$\frac{\text{BIS}}{\text{EVAL}} : \frac{-1.538s^3 + 0.2385s^2 - 0.1168s + 0.003513}{s^4 + 0.1798s^3 + 0.1461s^2 + 0.01316s + 0.0001649} \tag{5.14}$$

$$\frac{\text{BIS}}{\text{LMA}} : \frac{-0.02081s^3 + 1.765s^2 + 0.1646s + 0.002294}{s^4 + 0.1798s^3 + 0.1461s^2 + 0.01316s + 0.0001649} \tag{5.15}$$

Keeping all the parameters discussed above the same, simulations were run with some disturbances to the patient. Figure 5.10 illustrates the effects these disturbances have on the BIS trajectory of Patient 1.

40
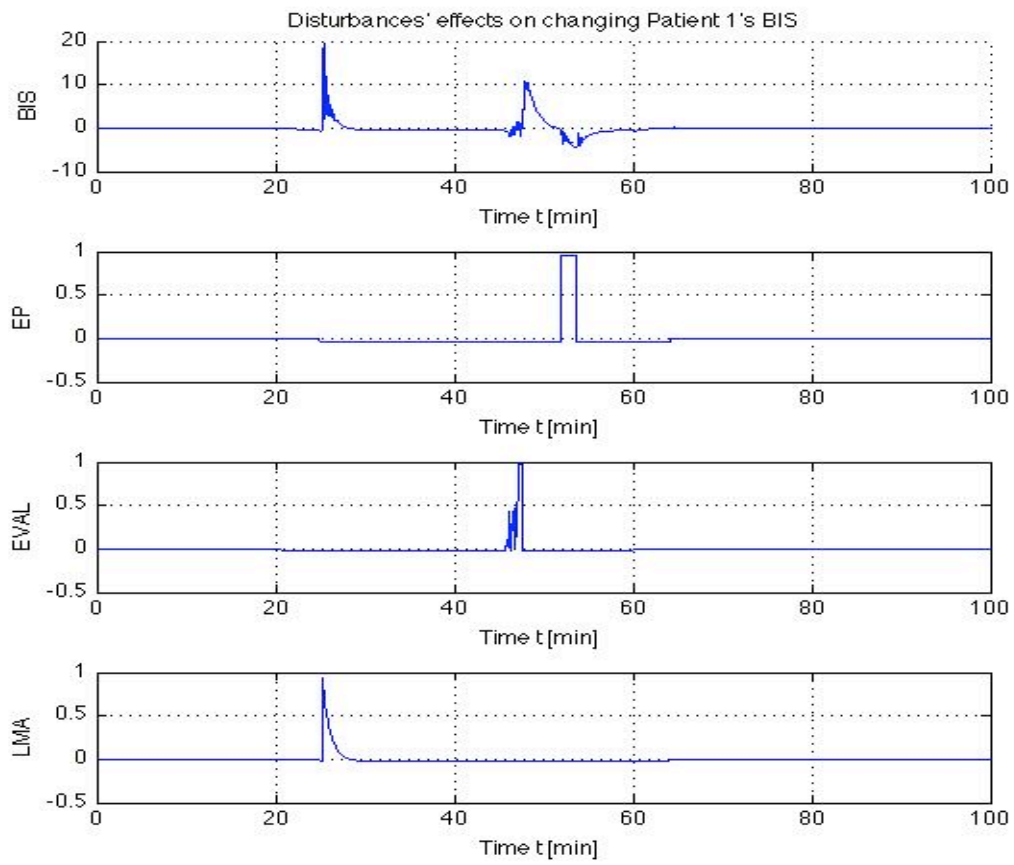
Figure 5.10: Disturbances' effect on Patient 1's BIS
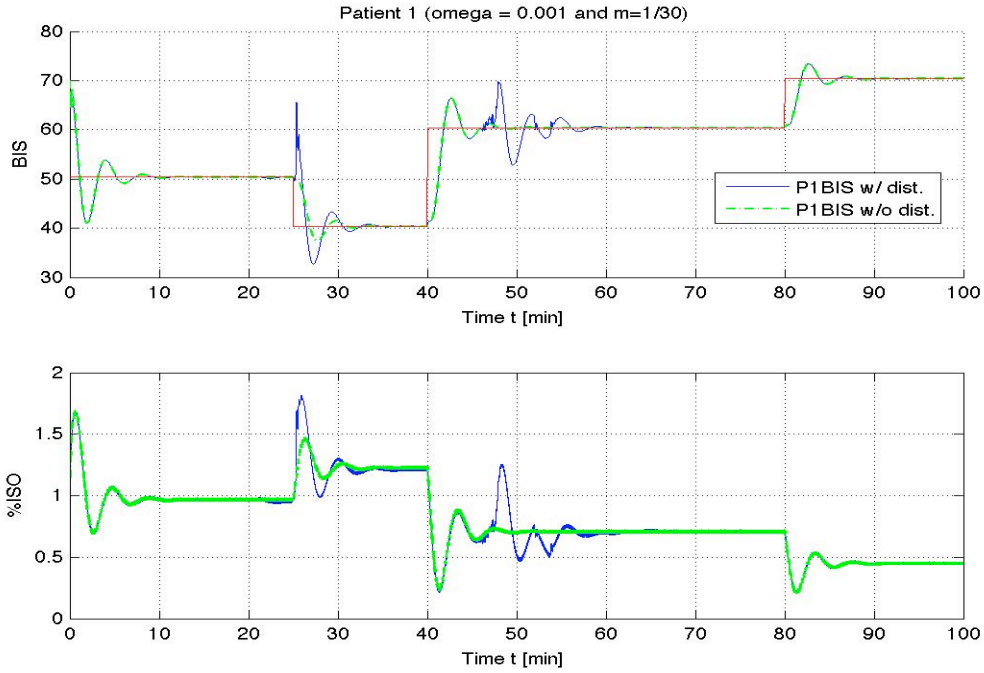
Figure 5.11: Patient 1 output feedback controller with $\Gamma = 50000$ w/disturbances

Figure 5.11 shows the closed-loop performance of the $\mathcal{L}_1$-adaptive controller used on Patient 1 when disturbances are introduced. The normalized residual error computed using (5.12) is 0.0022 and the amount of isoflurane used is 2.409 liters. For both simulations with and without disturbances, the system has a non-zero initial condition which leads to an exponentially decaying term in the control and system state signal; this does not affect the performance of the system throughout. In Tables 5.5 and 5.6 we present some results demonstrating various controller designs' performance in terms of residual error terms and isoflurane consumption with disturbances acting on the patient.

Table 5.5: Patient 1 with disturbances normalized residual errors: BIS tracking performance

ω/m	1/15	1/30	1/60
0.002	N/A	0.0041	0.0065
0.001	0.0059	0.0022	0.0024
0.0005	0.0048	0.0045	0.0028
0.0001	0.0075	0.0080	0.0075

www.manaraa.com

Table 5.6: Patient 1 with disturbances control effort: Isoflurane use in liters

ω/m	1/15	1/30	1/60
0.002	N/A	2.401	2.409
0.001	2.401	2.404	2.410
0.0005	2.406	2.409	2.416
0.0001	2.453	2.456	2.462

5.2.1.3 MAP performance, with disturbances

The mean arterial pressure is not an output we would like to have track a specific reference trajectory. Instead, during surgery, there is a range the anesthesiologist would like the patient's MAP to remain in. In our study, we would like the MAP to stay in the 60 to 110 mmHg range. The MAP transfer functions for the model of *Patient 1-sedated state* are:

$$\text{ISO to MAP: } \frac{-0.04938}{s + 0.005276} \tag{5.16}$$

$$\text{EP to MAP: } \frac{0.0126}{s + 0.005276} \tag{5.17}$$

$$\text{EVAL to MAP: } \frac{-0.1595}{s + 0.005276} \tag{5.18}$$

$$\text{LMA to MAP: } \frac{0.4991}{s + 0.005276} \tag{5.19}$$

43
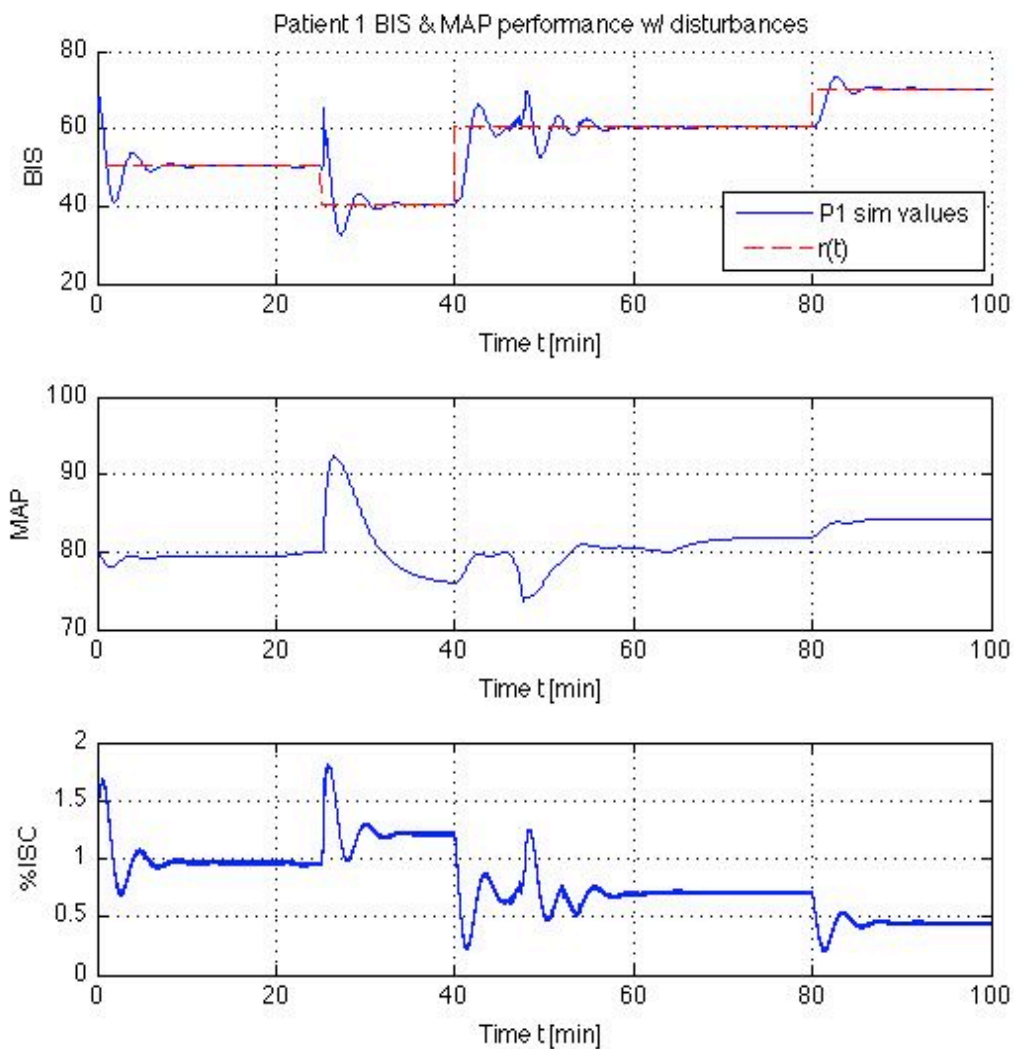
Figure 5.12: BIS reference tracking: Patient 1 BIS and MAP performance

Figure 5.12 illustrates the BIS and MAP performance achieved when the $\mathscr{L}_1$-adaptive output feedback controller with the previously detailed parameters is applied to Patient 1. Clearly, Patient 1's MAP is well within our desired range throughout the entire simulation.

5.2.1.4 Robustness to Inter-patient Variability

This controller design was then simulated using the models for Patients 2, 3, 5, 6, and 7 to evaluate inter-patient design robustness. That is, we applied the controller designed for Patient 1 on Patients 2, 3, 5, 6, and 7. In general, the patients' BIS levels tracked the desired reference profile closely, regardless of whether the applied controller was that specifically designed for them or not. Tables 5.7 and 5.8 detail the tracking performance and control effort when the controller designed for Patient 1 is applied to the models of Patients 2, 3, 5, 6, and 7.

Table 5.7: Normalized residual errors with Patient 1 control

Patient/Controller	1
2	0.0062
3	0.0022
5	0.0016
6	0.0036
7	0.0064

Table 5.8: Isoflurane use in liters with Patient 1 control

Patient/Controller	1
2	2.859
3	2.232
5	2.400
6	3.292
7	2.147

45
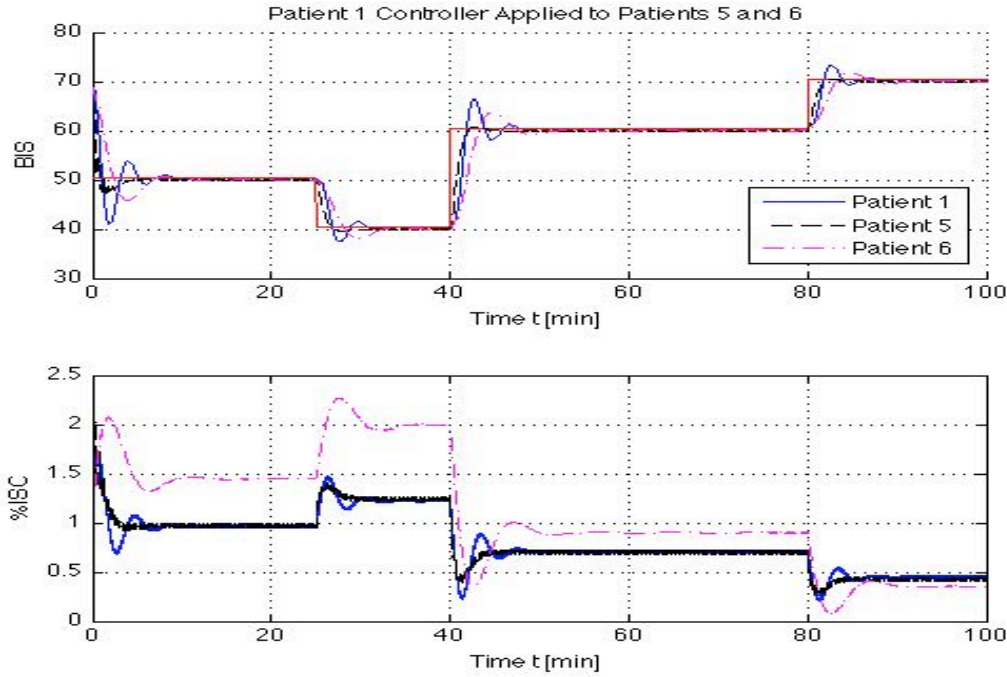
Figure 5.13: BIS reference tracking: Patient 1 robustness to inter-patient variability

Figure 5.13 illustrates how robust this controller design is to inter-patient variability by comparing the performance when Patient 1's controller is applied to Patients 1, 5, and 6. It is important to note that the controller is *exactly* the same between the patients.

5.2.2   Patient 2 $\mathscr{L}_1$-Adaptive Control-Output Feedback

As previously discussed, *black box* models were constructed by using the MAT-LAB *n4sid* command on partitioned patient data, and a MATLAB m-file was made to identify the values for all parameters required to simulate the controlled system detailed in (2.4), and for the $\mathscr{L}_1$-adaptive controller given by (3.25), (3.26), and (3.29). Simulink was then used to simulate the closed-loop system and obtain the performance results. The model parameters for *Patient 2-sedated state* are:

$$
A = \begin{bmatrix}
-0.0026 & -0.0279 & -0.0025 & -0.0244 \\
0.0177 & -0.0864 & 0.0056 & -0.2098 \\
-0.0008 & -0.0104 & 0.0066 & 0.1733 \\
0.0013 & -0.0431 & -0.1563 & -0.1275
\end{bmatrix},
$$

46

$$B = \begin{bmatrix} 0.0003 \\ 0.0810 \\ 0.0015 \\ 0.0015 \end{bmatrix},$$

$$C = \begin{bmatrix} 237.2070 \\ -11.6216 \\ 0.5634 \\ -2.9660 \end{bmatrix} \tag{5.20}$$

5.2.2.1 Tracking, no disturbances

The transfer function obtained from the model given in (5.20) is:

$$\frac{\text{BIS}}{\text{ISO}} : \frac{0.05594s^3 + 0.001404s^2 + 0.002921s - 0.0002398}{s^4 + 0.2099s^3 + 0.02877s^2 + 0.002828s + 2.01e - 05} \tag{5.21}$$

As in Patient 1's case, it is assumed the ISO to BIS transfer function is strictly proper. We set $P = 1$, $\Delta = 100$ and $\Gamma = 50000$. The $\mathscr{L}_1$-adaptive controller given by (3.25), (3.26), and (3.29) is used to track a given reference BIS trajectory, $r(t)$. If, using (2.7), we set $m = 1/60$ and set $\omega = 0.002$, using (3.9), (3.10) is strictly proper and BIBO stable as required. In Figure 5.14, we show the simulation results for the $\mathscr{L}_1$-adaptive controller applied to the dynamic response model for Patient 2.

As can be seen in Figure 5.14, using $\mathscr{L}_1$-adaptive output feedback control design previously discussed, we are able to achieve our desired tracking performance quickly with little tracking error. Using this particular controller design, we achieve a normalized error residual, given in (5.12), of 0.0050. The approximate isoflurane used for Patient 2 to obtain the achieved performance is 2.852 liters. Tables 5.9 and 5.10 detail the achieved performance with various controller designs for Patient 2.

47
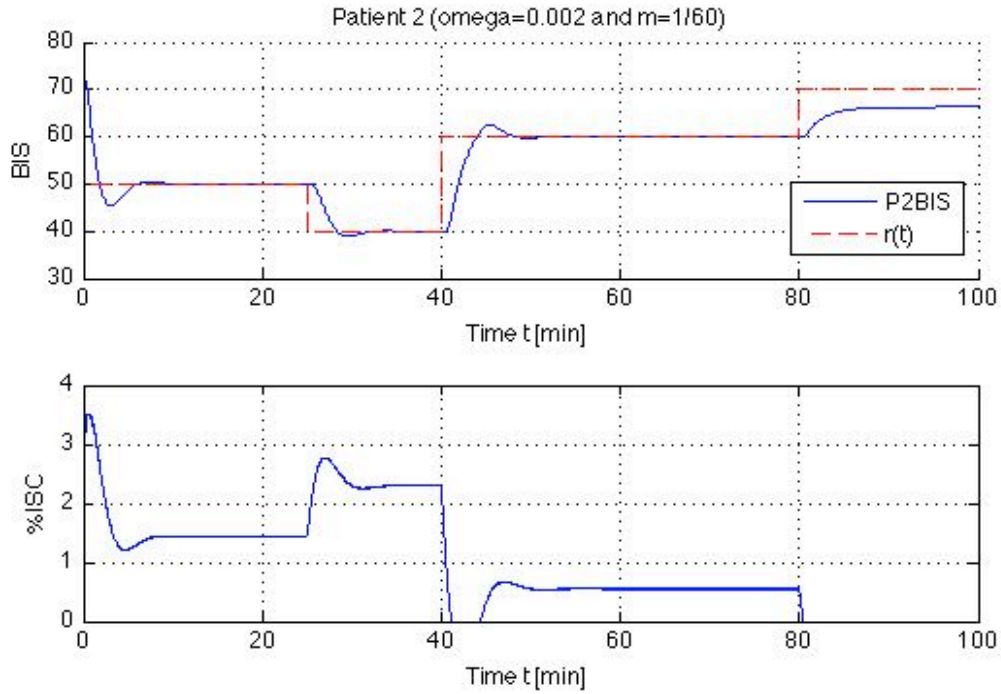
Figure 5.14: BIS reference tracking: Patient 2 output feedback controller with
$\Gamma = 50000$

Table 5.9: Patient 2 normalized residual errors: BIS tracking performance

ω/m	1/15	1/30	1/60
0.01	N/A	0.0055	0.0052
0.008	0.0064	0.0056	0.0055
0.004	0.0064	0.0050	0.0053
0.002	0.0063	0.0055	0.0050
0.001	0.0073	0.0062	0.0060
0.0005	0.0077	0.0077	0.0076
0.0001	0.0147	0.0146	0.0151

Table 5.10: Patient 2 control effort: Isoflurane use in liters

ω/m	1/15	1/30	1/60
0.01	N/A	2.856	2.840
0.008	2.863	2.858	2.843
0.004	2.867	2.859	2.847
0.002	2.860	2.857	2.852
0.001	2.860	2.859	2.861
0.0005	2.879	2.881	2.885
0.0001	3.067	3.069	3.074

5.2.2.2 Tracking, with disturbances

Just as before, the EP, EVAL, and LMA are treated as disturbance inputs to our ISO/BIS system. The transfer functions for the disturbances model of *Patient 2-sedated state* are:

$$\frac{\text{BIS}}{\text{EP}} : \frac{-0.6242s^3 - 0.03338s^2 - 0.002487s - 0.0004854}{s^4 + 0.2099s^3 + 0.02877s^2 + 0.002828s + 2.01e - 05} \tag{5.22}$$

$$\frac{\text{BIS}}{\text{EVAL}} : \frac{0.135s^3 + 0.1622s^2 - 0.0002242s + 0.0009576}{s^4 + 0.2099s^3 + 0.02877s^2 + 0.002828s + 2.01e - 05} \tag{5.23}$$

$$\frac{\text{BIS}}{\text{LMA}} : \frac{-0.87111s^3 + 0.1013s^2 - 0.008009s + 0.0003772}{s^4 + 0.2099s^3 + 0.02877s^2 + 0.002828s + 2.01e - 05} \tag{5.24}$$

The disturbances' effects on the BIS output are exactly the same as those of Patient 1. This was done for comparison and analysis purposes. Maintaining the same control design as discussed above, simulations were run with disturbance inputs.

Figure 5.15 shows the closed-loop performance of the $\mathcal{L}_1$-adaptive controller designed for Patient 2 when disturbances are introduced to the system. This particular system results in a calculated normalized residual error with a value of 0.0060. Approximate isoflurane use for Patient 2 is 2.852 liters. For both simulations with and without disturbances, the system has a non-zero initial condition which leads to an exponentially decaying term in the control and system state sig-
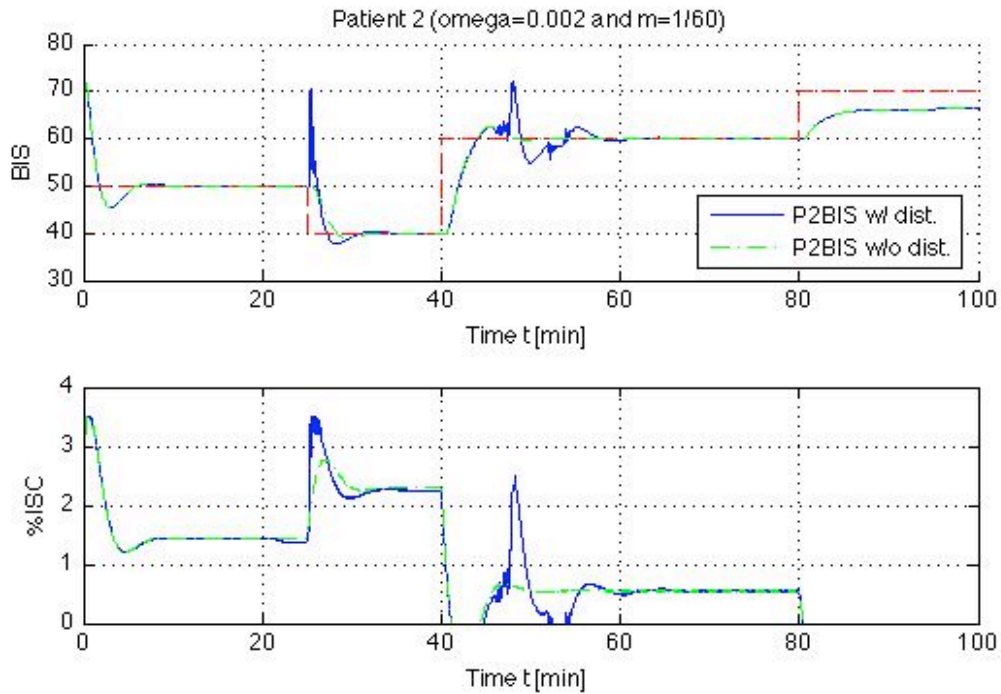
Figure 5.15: BIS reference tracking: Patient 2 output feedback controller with $\Gamma = 50000$ w/ disturbances

nal; this does not affect the performance of the system throughout. In Tables 5.11 and 5.12 we have some results demonstrating how different controller designs perform with disturbances.

Table 5.11: Patient 2 with disturbances normalized residual errors: BIS tracking performance

ω/m	1/15	1/30	1/60
0.01	N/A	0.0071	N/A
0.008	N/A	0.0071	0.0063
0.004	0.0087	0.0070	0.0057
0.002	0.0086	0.0076	0.0060
0.001	0.0085	0.0084	0.0071
0.0005	0.0093	0.0092	0.0086
0.0001	0.0156	0.0159	0.0159

Table 5.12: Patient 2 with disturbances control effort: Isoflurane use in liters

ω/m	1/15	1/30	1/60
0.01	N/A	2.862	N/A
0.008	N/A	2.863	2.837
0.004	2.891	2.866	2.844
0.002	2.867	2.862	2.852
0.001	2.861	2.859	2.860
0.0005	2.879	2.881	2.884
0.0001	3.068	3.070	3.074

5.2.2.3 MAP performance, with disturbances

Unlike the BIS, the mean arterial pressure is not an output we would like to have track a specific reference trajectory. Instead, during surgery, there is an ideal range the anesthesiologist would like the patient's MAP to remain in. In our study, we would like the MAP to stay in the 60 to 110 mmHg range. The MAP transfer functions for the model of *Patient 2-sedated state* are:

$$\text{ISO to MAP: } \frac{0.09125s^2 - 0.003802s + 0.001702}{s^3 + 0.06737s^2 + 0.1591s + 0.0003529} \tag{5.25}$$

$$\text{EP to MAP: } \frac{-0.0304s^2 - 0.01384s - 0.002925}{s^3 + 0.06737s^2 + 0.1591s + 0.0003529} \tag{5.26}$$

$$\text{EVAL to MAP: } \frac{-0.02734s^2 - 0.01166s - 0.002819}{s^3 + 0.06737s^2 + 0.1591s + 0.0003529} \tag{5.27}$$

$$\text{LMA to MAP: } \frac{-0.1536s^2 - 0.1712s - 0.0122}{s^3 + 0.06737s^2 + 0.1591s + 0.0003529} \tag{5.28}$$

Figure 5.16 illustrates the BIS and MAP performance achieved when the $\mathcal{L}_1$-adaptive output feedback controller with the previously detailed parameters is applied to Patient 2. Clearly, Patient 2's MAP is well within our desired range throughout the entire simulation.
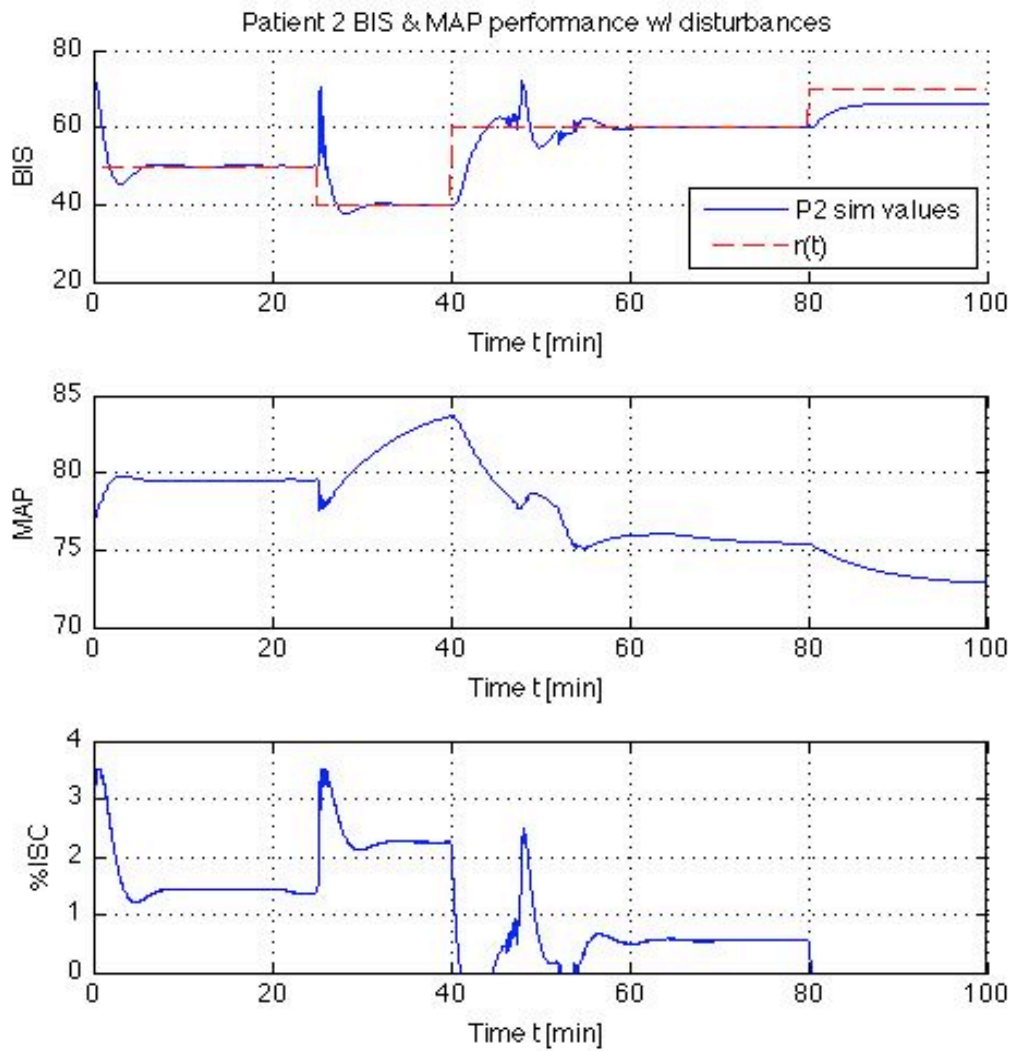
Figure 5.16: BIS reference tracking: Patient 2 BIS and MAP performance

5.2.2.4 Robustness to Inter-patient Variability

This controller design was then simulated using the models for Patients 1, 3, 5, 6, and 7 to evaluate inter-patient design robustness. That is, we applied the controller designed for Patient 2 on Patients 1, 3, 5, 6, and 7. Except for Patient 5, the patients' BIS levels tracked the desired reference profile closely, regardless of whether the applied controller was that specifically designed for them or not. Tables 5.13 and 5.14 detail the tracking performance and control effort when the controller designed for Patient 2 is applied to the models of Patients 1, 3, 5, 6, and 7.

Table 5.13: Normalized residual errors with Patient 2 control

Patient/Controller	2
1	0.0047
3	0.0030
5	N/A
6	0.0027
7	0.0029

Table 5.14: Isoflurane use in liters with Patient 2 control

Patient/Controller	2
1	2.408
3	2.224
5	N/A
6	3.292
7	2.141

5.2.3   Patient 3 $\mathscr{L}_1$-Adaptive Control-Output Feedback

For Patient 3, models were constructed through the MATLAB *n4sid* command on partitioned patient data. A MATLAB m-file was created to identify the values for all parameters required for the system in (2.4), and for the $\mathscr{L}_1$-adaptive controller given by (3.25), (3.26), and (3.29). Simulink was then used to simulate the closed-loop systems.

54

The model parameters for *Patient 3-sedated state* are:

$$A = \begin{bmatrix} -0.0010 & -0.0276 & 0.0235 & 0.0076 \\ 0.0088 & -0.0494 & 0.1102 & 0.0505 \\ 0.0018 & 0.0445 & -0.1327 & -0.3432 \\ 0.0004 & -0.0125 & 0.2824 & 0.0256 \end{bmatrix},$$

$$B = \begin{bmatrix} -0.0082 \\ -0.0185 \\ -0.0166 \\ -0.0381 \end{bmatrix},$$

$$C = \begin{bmatrix} 173.0765 \\ -9.0676 \\ 2.5768 \\ -2.4800 \end{bmatrix} \tag{5.29}$$

5.2.3.1 Tracking, no disturbances

The transfer function for the model of *Patient 3-sedated state* are:

$$\frac{\text{BIS}}{\text{ISO}} : \frac{-1.2s^3 - 0.1397s^2 - 0.06413s - 0.0002461}{s^4 + 0.1575s^3 + 0.09489s^2 + 0.003838s + 2.527e - 05} \tag{5.30}$$

Like Patients 2 and 3, it is assumed the ISO to BIS transfer function is strictly proper. We set $P = 1$ and conservatively set $\Delta = 100$ and $\Gamma = 50000$. The $\mathscr{L}_1$-adaptive controller given by (3.25), (3.26), and (3.29) is used to track a given reference BIS trajectory, $r(t)$. Using (2.7) and (3.9), we set $m = 1/15$ and for set $\omega = 0.008$, respectively. Using this, we can show that (3.10) is strictly proper and BIBO stable as required. Simulation results for the $\mathscr{L}_1$-adaptive controller applied to the dynamic response model for Patient 3 can be seen in Figure 5.17.

As can be seen in Figure 5.17, the $\mathscr{L}_1$-adaptive output feedback control design achieves the desired tracking performance quickly with little tracking error. This particular design when used to control Patient 3's BIS trajectory achieves a normalized error residual of 0.0015. The approximate isoflurane amount required to achieve this tracking performance for Patient 3 is 2.197 liters.

55

Table 5.15: Patient 3 normalized residual errors: BIS tracking performance

ω/m	1/15	1/30	1/60
0.16	8.662e-04	0.0014	0.0030
0.08	9.866e-04	0.0011	0.0019
0.04	9.944e-04	0.0011	0.0017
0.02	0.0012	0.0013	0.0017
0.01	0.0014	0.0016	0.0018
0.008	0.0015	0.0016	0.0019
0.004	0.0013	0.0020	0.0023
0.002	0.0048	0.0024	0.0030
0.001	0.0053	0.0022	0.0039
0.0005	0.0066	0.0062	0.0053
0.0001	0.0128	0.0128	0.0133

Table 5.16: Patient 3 control effort: Isoflurane use in liters

ω/m	1/15	1/30	1/60
0.16	2.278	2.365	2.536
0.08	2.218	2.250	2.328
0.04	2.198	2.212	2.241
0.02	2.194	2.200	2.216
0.01	2.196	2.200	2.211
0.008	2.197	2.202	2.211
0.004	2.201	2.206	2.216
0.002	2.210	2.215	2.224
0.001	2.228	2.232	2.242
0.0005	2.263	2.268	2.277
0.0001	2.481	2.483	2.487
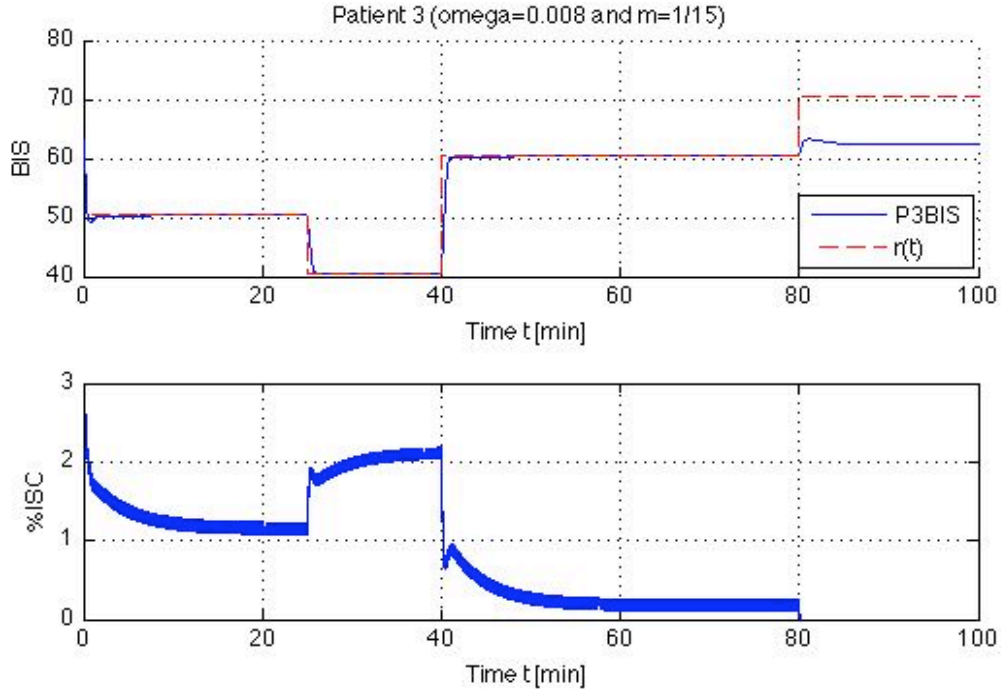
Figure 5.17: BIS reference tracking: Patient 3 output feedback controller with $\Gamma = 50000$

5.2.3.2 Tracking, with disturbances

Again, the EP/BIS, EVAL/BIS, and LMA/BIS are treated as disturbances to our ISO/BIS system. The transfer functions for the disturbances model of *Patient 3-sedated state* are:

$$\frac{\text{BIS}}{\text{EP}} : \frac{0.3388s^3 - 0.279s^2 + 0.01427s - 0.0002118}{s^4 + 0.1575s^3 + 0.09489s^2 + 0.003838s + 2.527e - 05} \tag{5.31}$$

$$\frac{\text{BIS}}{\text{EVAL}} : \frac{-0.6152s^3 + 0.1421s^2 - 0.03961s + 0.002399}{s^4 + 0.1575s^3 + 0.09489s^2 + 0.003838s + 2.527e - 05} \tag{5.32}$$

$$\frac{\text{BIS}}{\text{LMA}} : \frac{2.315s^3 + 1.921s^2 + 0.1231s + 0.001609}{s^4 + 0.1575s^3 + 0.09489s^2 + 0.003838s + 2.527e - 05} \tag{5.33}$$

As before, the disturbances effects on the BIS output are exactly the same as those of Patient 1. This was done for comparison and analysis purposes. Main-

57

taining the same control design as discussed above, simulations were run with disturbance affecting Patient 3's BIS trajectory.
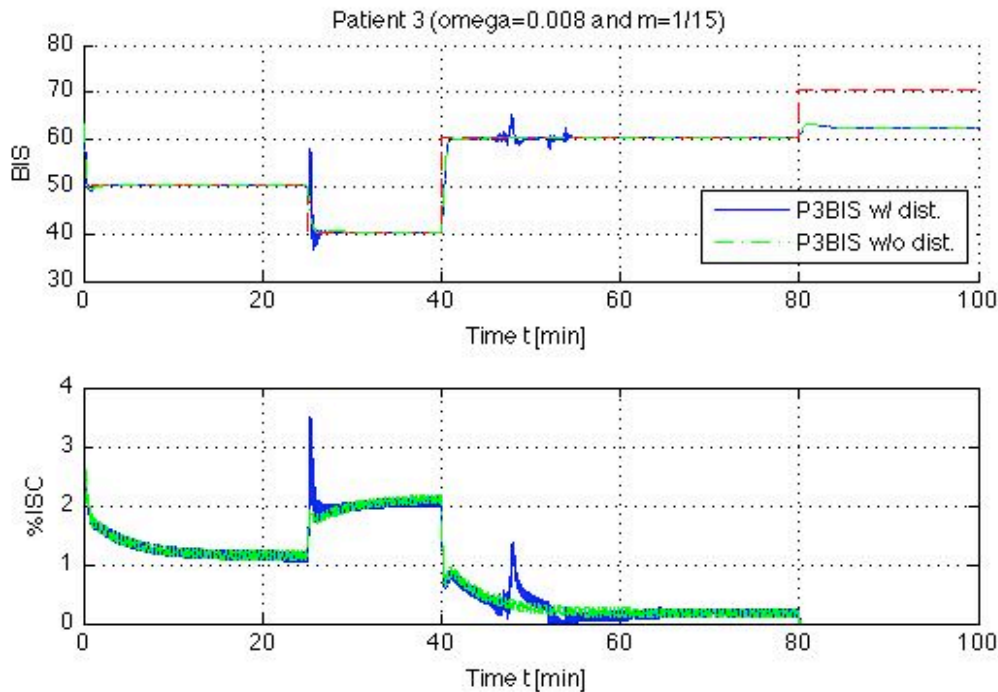
Figure 5.18: BIS reference tracking: Patient 3 output feedback controller with $\Gamma = 50000$ w/ disturbances

Figure 5.18 shows the closed-loop performance of the $\mathcal{L}_1$-adaptive controller used on Patient 3 when disturbances are introduced. This particular system results in a calculated normalized residual error with a value of 0.0016. Approximate isoflurane use for Patient 3 is 2.197 liters. Note, for both simulations with and without disturbances, the system has a non-zero initial condition which leads to an exponentially decaying term in the control and system state signal; this does not affect the performance of the system throughout.

In Tables 5.17 and 5.18 we have some results of various design configurations for dealing with disturbances.

5.2.3.3 MAP performance, with disturbances

We now examine how the $\mathcal{L}_1$-adaptive output feedback controller previously discussed affects the mean arterial pressure of Patient 3. Again, there is a desired range the anesthesiologist would like the patient's MAP to remain in. In our study,

58

Table 5.17: Patient 3 with disturbances normalized residual errors: BIS tracking performance

ω/m	1/15	1/30	1/60
0.16	9.2918e-04	0.0015	0.0032
0.08	0.0010	0.0012	0.0019
0.04	0.0010	0.0011	0.0017
0.02	0.0013	0.0014	0.0017
0.01	0.0015	0.0016	0.0018
0.008	0.0016	0.0016	0.0019
0.004	0.0017	0.0021	0.0023
0.002	0.0049	0.0030	0.0030
0.001	0.0056	0.0026	0.0039
0.0005	0.0069	0.0069	0.0054
0.0001	0.0135	0.0135	0.0138

Table 5.18: Patient 3 with disturbances control effort: Isoflurane use in liters

ω/m	1/15	1/30	1/60
0.16	2.276	2.360	2.523
0.08	2.218	2.248	2.323
0.04	2.199	2.212	2.241
0.02	2.195	2.201	2.218
0.01	2.196	2.201	2.213
0.008	2.197	2.202	2.213
0.004	2.201	2.206	2.217
0.002	2.210	2.215	2.225
0.001	2.228	2.233	2.242
0.0005	2.264	2.268	2.277
0.0001	2.484	2.486	2.489

we would like the MAP to stay in the 60 to 110 mmHg range. The MAP transfer functions for the model of *Patient 3-sedated state* are:

$$\text{ISO to MAP: } \frac{1.03s^3 + 0.09138s^2 + 0.01881s + 0.0004485}{s^4 + 0.3259s^3 + 0.09716s^2 + 0.003421s + 3.747e - 07} \tag{5.34}$$

$$\text{EP to MAP: } \frac{-0.06771s^3 + 0.05368s^2 - 0.002627s + 0.0002745}{s^4 + 0.3259s^3 + 0.09716s^2 + 0.003421s + 3.747e - 07} \tag{5.35}$$

$$\text{EVAL to MAP: } \frac{0.0358s^3 + 0.01821s^2 + 0.002357s + 0.0003508}{s^4 + 0.3259s^3 + 0.09716s^2 + 0.003421s + 3.747e - 07} \tag{5.36}$$

$$\text{LMA to MAP: } \frac{-0.4936s^3 + 0.8605s^2 + 0.01085s + 0.001188}{s^4 + 0.3259s^3 + 0.09716s^2 + 0.003421s + 3.747e - 07} \tag{5.37}$$

Figure 5.19 illustrates the BIS and MAP performance achieved when the $\mathscr{L}_1$-adaptive output feedback controller with the previously detailed parameters is applied to Patient 3. Clearly, Patient 3's MAP is well within our desired range throughout the entire simulation.

60
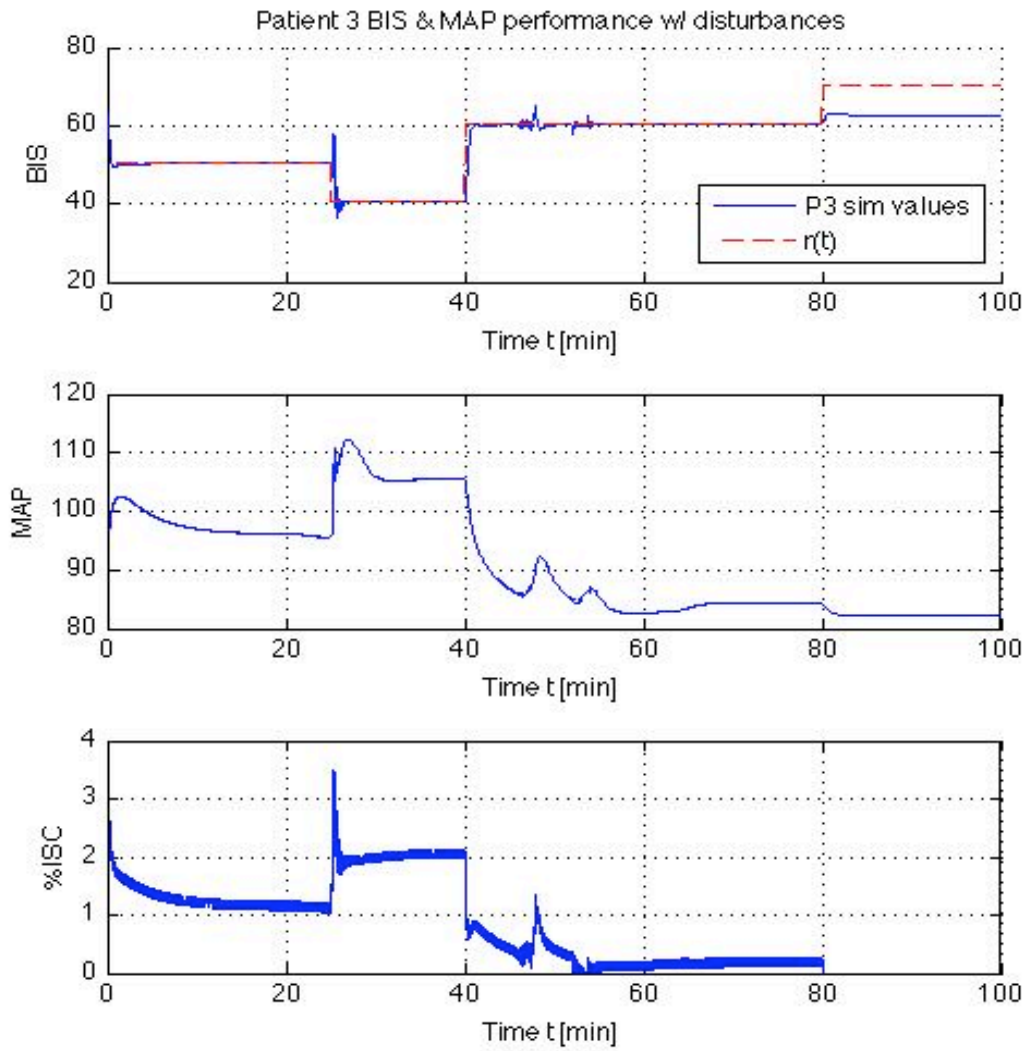
Figure 5.19: BIS reference tracking: Patient 3 BIS and MAP performance

5.2.3.4 Robustness to Inter-patient Variability

Similarly to our examination of Patient 1's controller, Patient 3's controller design was then simulated using the models for Patients 1, 2, 5, 6, and 7 to evaluate inter-patient design robustness. That is, we applied the controller designed for Patient 3 on Patients 1, 2, 5, 6, and 7. When Patients 2, 6, and 7 use the Patient 3 controller, their BIS levels tracked the desired reference profile closely. Tables 5.19 and 5.20 detail the tracking performance and control effort when the controller designed for Patient 3 is applied to the models of Patients 1, 2, 5, 6, and 7. Note that Patients 1 and 5 resulted in unstable $H(s)$ when Patient 3's controller is applied. This is due to the fact that Patient 3's controller is more aggressive than most controllers investigated and results in an unstable $H(s)$, given in (3.10).

Table 5.19: Normalized residual errors with Patient 3 Control

Patient/Controller	3
1	N/A
2	0.0064
5	N/A
6	0.0024
7	0.0044

Table 5.20: Isoflurane use in liters with Patient 3 control

Patient/Controller	3
1	N/A
2	2.863
5	N/A
6	3.288
7	2.143

5.2.4   Patient 5 $\mathscr{L}_1$-Adaptive Control-Output Feedback

For Patient 5, we constructed *black box* models through the use of the MATLAB *n4sid* command on partitioned patient data. A MATLAB m-file was created and

then used to calculate the values for all parameters required for the system in (2.4), and for the $\mathscr{L}_1$-adaptive controller given by (3.25), (3.26), and (3.29). Simulink was then used to simulate the closed-loop systems.

The model parameters for *Patient 5-sedated state* are:

$$A = \begin{bmatrix} -0.0553 & 0.0037 & 0.0329 & -0.0265 \\ 0.6119 & -0.5097 & 0.1335 & 0.2688 \\ 0.9460 & -0.6811 & -0.0161 & 0.5125 \\ -0.6282 & 0.3854 & -0.1738 & -0.1729 \end{bmatrix},$$

$$B = \begin{bmatrix} -0.0070 \\ 0.0833 \\ 0.1311 \\ -0.0873 \end{bmatrix},$$

$$C = \begin{bmatrix} 139.8580 \\ -9.8886 \\ 0.4684 \\ -0.5230 \end{bmatrix} \tag{5.38}$$

5.2.4.1 Tracking, no disturbances

The transfer function for the model of *Patient 5-sedated state* are:

$$\frac{\text{BIS}}{\text{ISO}} : \frac{-2.221s^3 + 0.2494s^2 - 0.09021s - 0.002341}{s^4 + 0.1496s^3 + 0.06726s^2 + 0.006355s + 5.131e - 05} \tag{5.39}$$

Again, it is assumed the ISO to BIS transfer function is strictly proper. We set $P = 1$, and conservatively set $\Delta = 100$ and $\Gamma = 50000$. The $\mathscr{L}_1$-adaptive controller given by (3.25), (3.26), and (3.29) is used to track a given reference BIS trajectory, $r(t)$. If, using (2.7), we set $m = 1/30$ and we set $\omega = 0.002$ for (3.9), we attain an $H(s)$ in (3.10) that is strictly proper and BIBO stable as required. Simulation results for the $\mathscr{L}_1$-adaptive controller applied to the dynamic response model for Patient 5 can be seen in Figure 5.20.

As can be seen in Figure 5.20, the $\mathscr{L}_1$-adaptive output feedback control design achieves the desired tracking performance quickly with little tracking error. Compared to other model and contol design pairs, this system has one of the smallest

63
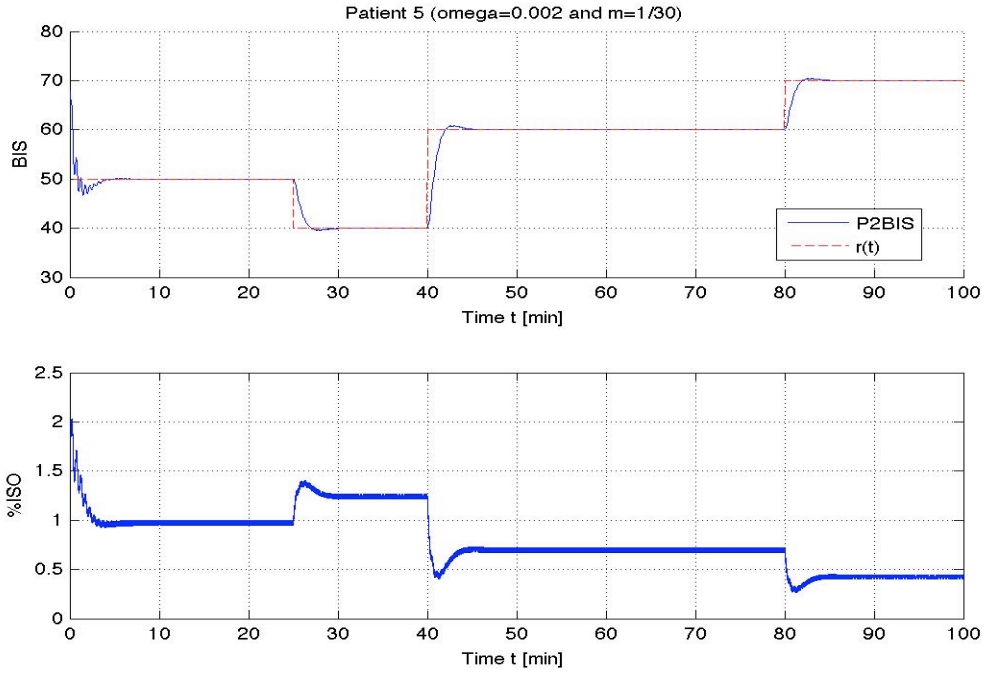
Figure 5.20: BIS reference tracking: Patient 5 output feedback controller with $\Gamma = 50000$

values of r_N (5.12) with a value of 0.0012. The approximate isoflurane use for Patient 5 is 2.397 liters; isoflurane use for a comparable BIS reference and simulation time resulting from our previous linear parameter-varying control studies was approximately 2.85 liters. The smaller amount of isoflurane used by the $\mathscr{L}_1$-adaptive output feedback controller would be better for the patients than what the LPV controller used.

Table 5.21: Patient 5 normalized residual errors: BIS tracking performance

ω/m	1/15	1/30	1/60
0.004	7.1646e-04	N/A	N/A
0.002	0.0015	0.0012	N/A
0.001	0.0021	0.0016	0.0020
0.0005	0.0029	0.0030	0.0027
0.0001	0.0073	0.0073	0.0074

Table 5.22: Patient 5 control effort: Isoflurane use in liters

ω/m	1/15	1/30	1/60
0.004	2.392	N/A	N/A
0.002	2.393	2.397	N/A
0.001	2.397	2.400	2.407
0.0005	2.403	2.406	2.413
0.0001	2.455	2.459	2.465

5.2.4.2 Tracking, with disturbances

For Patient 5, the EP/BIS, EVAL/BIS, and LMA/BIS, as with the others in our patient set, are treated as disturbances to our ISO/BIS system. The transfer functions for the disturbances model of *Patient 5-sedated state* are:

$$\frac{\text{BIS}}{\text{EP}} : \frac{-0.2798s^3 - 0.3321s^2 - 0.01559s - 0.0008389}{s^4 + 0.1798s^3 + 0.1461s^2 + 0.01316s + 0.0001649} \quad (5.40)$$

$$\frac{\text{BIS}}{\text{EVAL}} : \frac{-1.538s^3 + 0.2385s^2 - 0.1168s + 0.003513}{s^4 + 0.1798s^3 + 0.1461s^2 + 0.01316s + 0.0001649} \quad (5.41)$$

$$\frac{\text{BIS}}{\text{LMA}} : \frac{-0.02081s^3 + 1.765s^2 + 0.1646s + 0.002294}{s^4 + 0.1798s^3 + 0.1461s^2 + 0.01316s + 0.0001649} \quad (5.42)$$

Note, the disturbances used are exactly the same as those in the Patient 1 simulation. This was done to subject the system to the same disturbances to allow for comparison and analysis purposes.

Keeping all the parameters discussed above the same, simulations were run with some disturbances to the patient. Figure 5.21 illustrates the effects these disturbances have on the BIS of Patient 5. Figure 5.22 shows the closed-loop performance of the $\mathscr{L}_1$-adaptive controller used on Patient 5 when disturbances are introduced. This particular system results in a calculated normalized residual error (5.12) with a value of 0.0012. The approximate isoflurane use for Patient 5 is 2.397 liters. For both simulations with and without disturbances, the system has a non-zero initial condition which leads to an exponentially decaying term in the control and system state signal; this does not affect the performance of the system throughout.
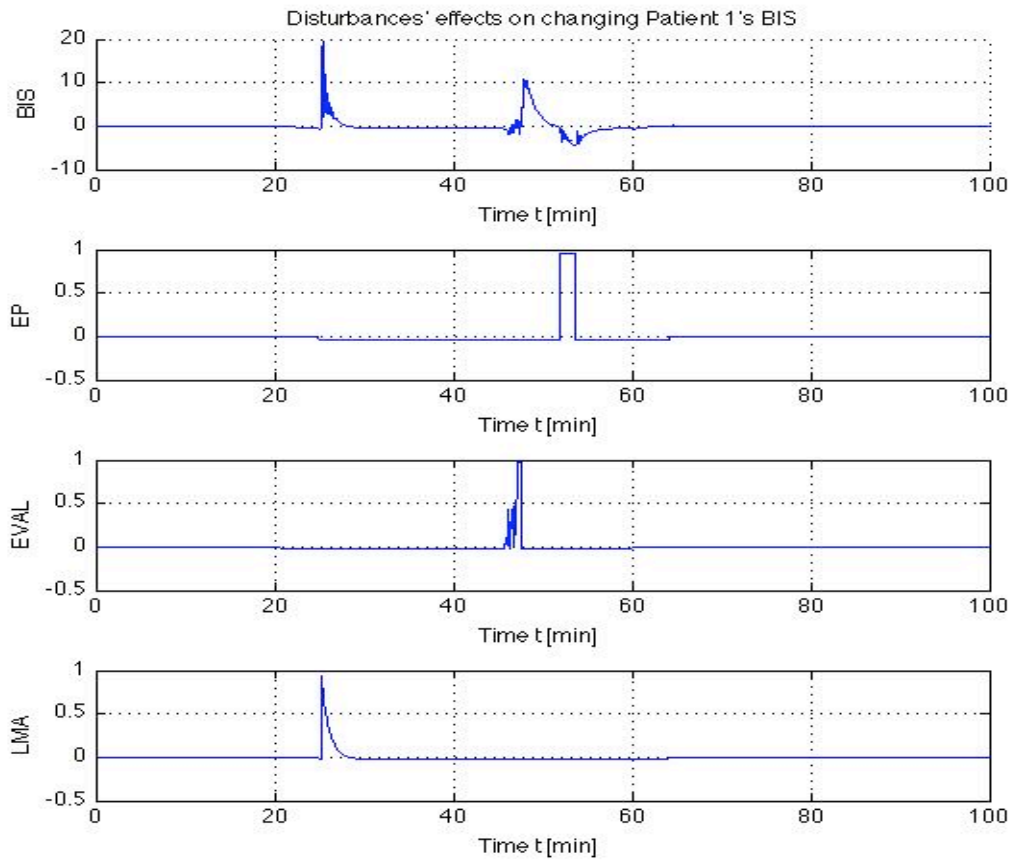
Figure 5.21: Disturbances effect on Patient 5's BIS
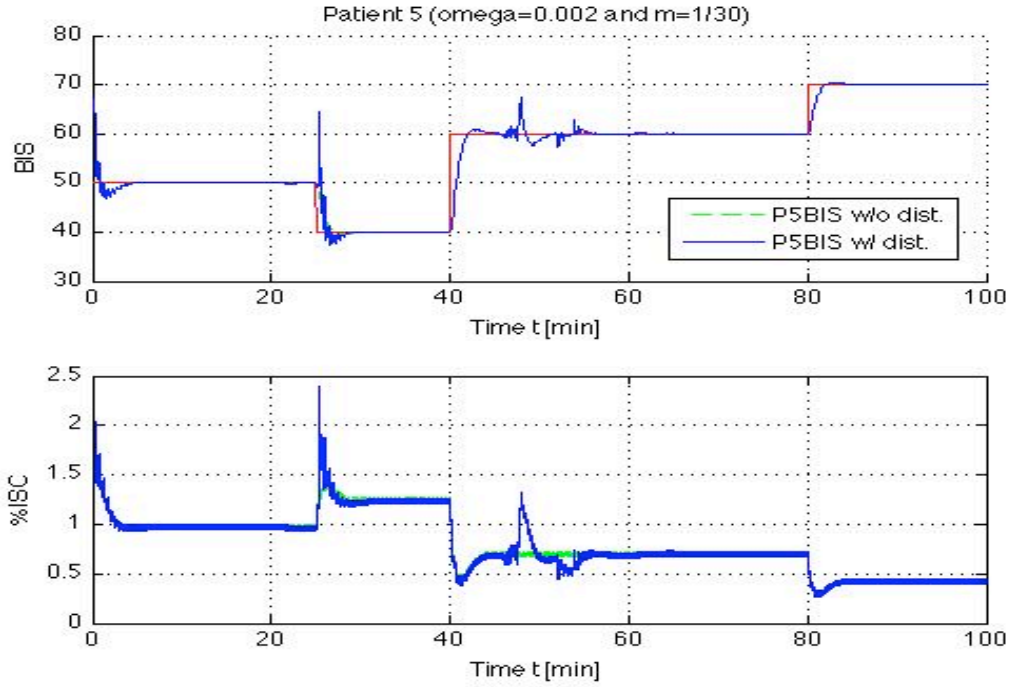
Figure 5.22: BIS reference tracking: Patient 5 output feedback controller with $\Gamma = 50000$ w/ disturbances

Tables 5.23 and 5.24 give tracking performance and control effort results with disturbances to the system achieved by different controller designs (i.e., $C(s)$ filters and $M(s)$ reference transfer function combinations).

Table 5.23: Patient 5 with disturbances normalized residual errors: BIS tracking performance

ω/m	1/15	1/30	1/60
0.004	8.2566e-4	N/A	N/A
0.002	0.0018	0.0012	N/A
0.001	0.0027	0.0017	0.0019
0.0005	0.0036	0.0038	0.0027
0.0001	0.0079	0.0082	0.0079

Table 5.24: Patient 5 with disturbances control effort: Isoflurane use in liters

ω/m	1/15	1/30	1/60
0.004	2.392	N/A	N/A
0.002	2.393	2.397	N/A
0.001	2.397	2.400	2.406
0.0005	2.403	2.406	2.413
0.0001	2.455	2.459	2.465

5.2.4.3 MAP performance, with disturbances

For Patient 5, the LMA and EP events occurred prior to the patient passing below the BIS threshold value of 70, so these events are not included in the data associated with the sedated state.

The MAP transfer functions for the model of *Patient 5-sedated state* are:

$$\text{ISO to MAP: } \frac{-0.073}{s + 0.005184} \tag{5.43}$$

$$\text{EVAL to MAP: } \frac{0.2056}{s + 0.005184} \tag{5.44}$$

68
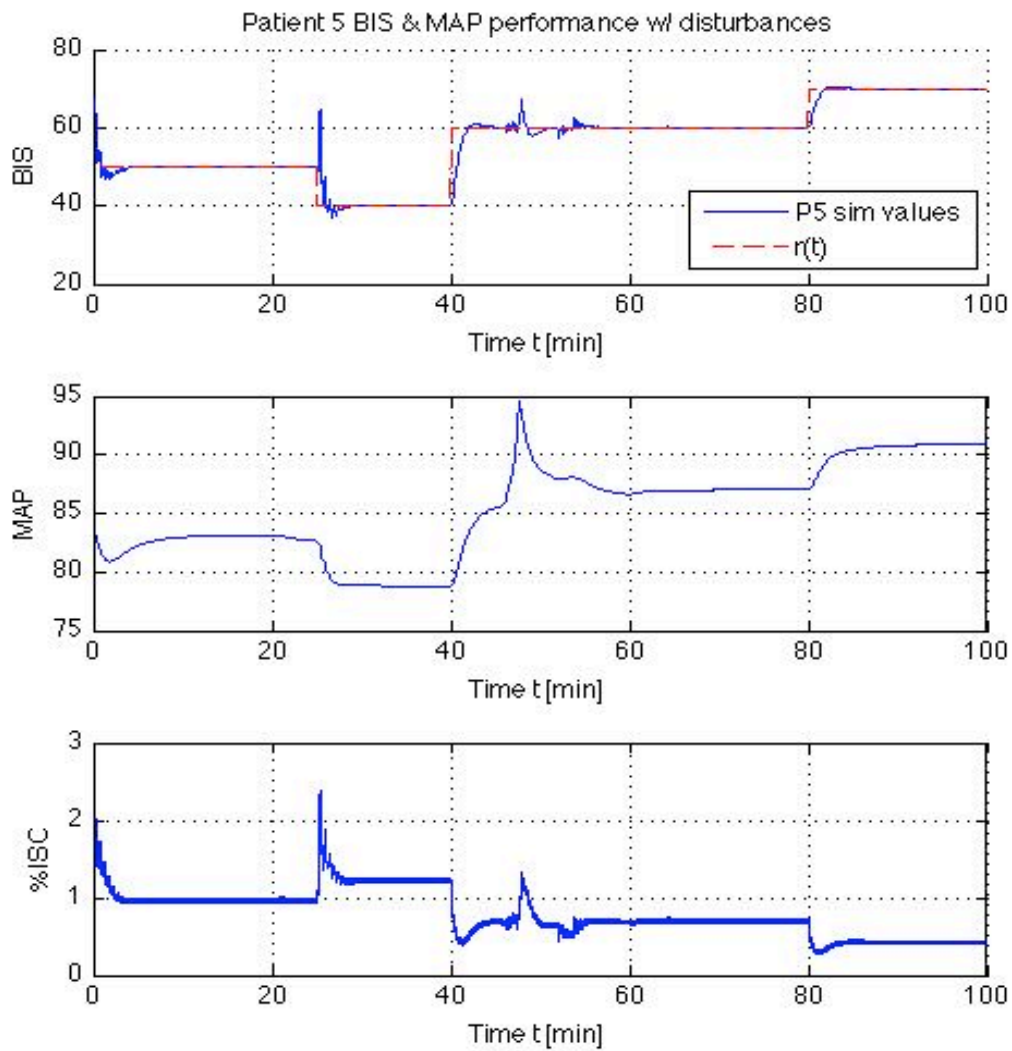
Figure 5.23: BIS reference tracking: Patient 5 BIS and MAP performance

Figure 5.23 shows the BIS and MAP simulated trajectories achieved when the $\mathcal{L}_1$-adaptive output feedback controller with the parameters used in Patient 5's controller as previously discussed when examining tracking performance with no disturbances. Patient 5's MAP clearly stays within the 60 to 110 mmHg range for the entirety of the simulation. Again, note the LMA and EP disturbances did not occur in the sedated data.

5.2.4.4 Robustness to Inter-patient Variability

Similarly to our examination of the other controllers robustness to interpatient variability, Patient 5's controller design was then simulated using the models for Patients 1, 2, 3, 6, and 7 to evaluate inter-patient design robustness. That is, we applied the controller designed for Patient 5 on Patients 1, 2, 3, 6, and 7. For every model in our patient set, the patients' BIS levels tracked the desired reference profile closely, regardless of whether the applied controller was that specifically designed for them or not. Tables 5.25 and 5.26 detail the tracking performance and control effort when the controller designed for Patient 5 is applied to the models of Patients 1, 2, 3, 6, and 7.

70

Table 5.25: Normalized residual errors with Patient 5 control, no disturbances

Patient/Controller	5
1	0.0027
2	0.0055
3	0.0024
6	0.0027
7	0.0055

Table 5.26: Isoflurane use in liters with Patient 5 control, no disturbances

Patient/Controller	5
1	2.401
2	2.857
3	2.215
6	3.285
7	2.137

71

Figure 5.24 illustrates how robust this controller design is to inter-patient variability. It is important to note that the controller is *exactly* the same between the patients.
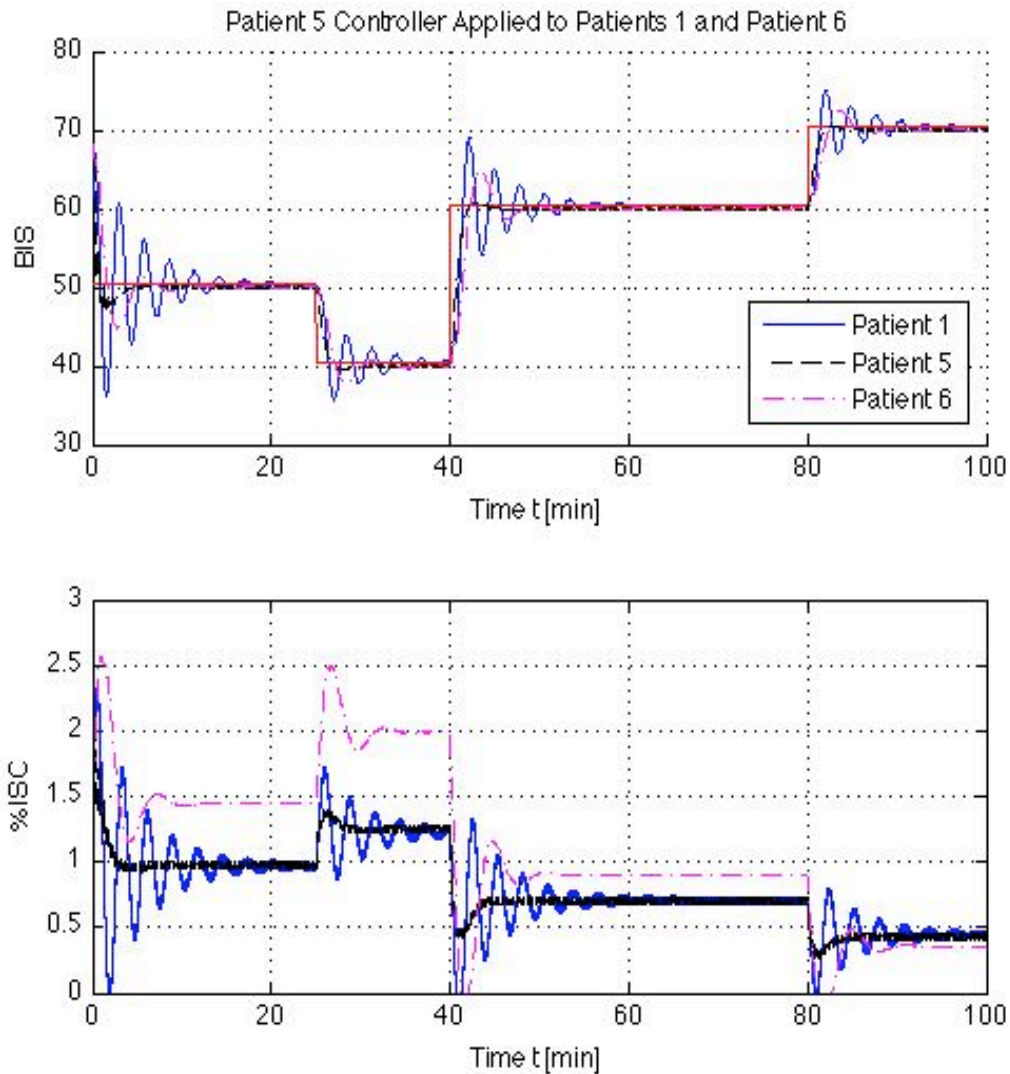
Figure 5.24: BIS reference tracking: Patient 5 robustness to inter-patient variability, no disturbances

5.2.5 Patient 6 $\mathscr{L}_1$-Adaptive Control-Output Feedback

For Patient 6, models were constructed through the MATLAB *n4sid* command on partitioned patient data. A MATLAB m-file was created to identify the values for

72

all parameters required for the system in (2.4), and for the $\mathscr{L}_1$-adaptive controller given by (3.25), (3.26), and (3.29). Simulink was then used to simulate the closed-loop systems.

The model parameters for *Patient 6-sedated state* are:

$$A = \begin{bmatrix} -0.0055 & 0.0096 & -0.0862 & 0.0357 \\ 0.0039 & -0.0137 & -0.1087 & 0.1693 \\ 0.0118 & -0.0580 & -0.1745 & 0.0230 \\ 0.0001 & -0.0924 & 0.1051 & -0.0576 \end{bmatrix},$$

$$B = \begin{bmatrix} 0.0025 \\ 0.0072 \\ 0.0267 \\ 0.0151 \end{bmatrix},$$

$$C = \begin{bmatrix} 137.6774 \\ -8.9751 \\ -5.8180 \\ 3.9340 \end{bmatrix}. \tag{5.45}$$

5.2.5.1 Tracking, no disturbances

The transfer function for the model of *Patient 6-sedated state* are:

$$\frac{\text{BIS}}{\text{ISO}} : \frac{0.1836s^3 - 0.1526s^2 + 0.003364s - 0.0005311}{s^4 + 0.2513s^3 + 0.02248s^2 + 0.003408s + 2.944e - 05} \tag{5.46}$$

Again, it is assumed the ISO to BIS transfer function is strictly proper. We conservatively set $P = 1$ and chose $\Delta = 100$ and $\Gamma = 50000$. The $\mathscr{L}_1$-adaptive controller given by (3.25), (3.26), and (3.29) is applied to track a desired sedated BIS trajectory, $r(t)$. If for $M(s)$ in (2.7) we set $m = 1/30$ and $C(s)$ in (3.9) set $\omega = 0.004$, we can show that (3.10) is strictly proper and BIBO stable as required. Simulation results for the $\mathscr{L}_1$-adaptive controller applied to the dynamic response model for Patient 6 can be seen in Figure 5.25.

As can be seen in Figure 5.25, the $\mathscr{L}_1$-adaptive output feedback control design achieves the desired tracking performance quickly with little tracking error. This particular design achieves a normalized error residual of 0.0021. The approximate

73
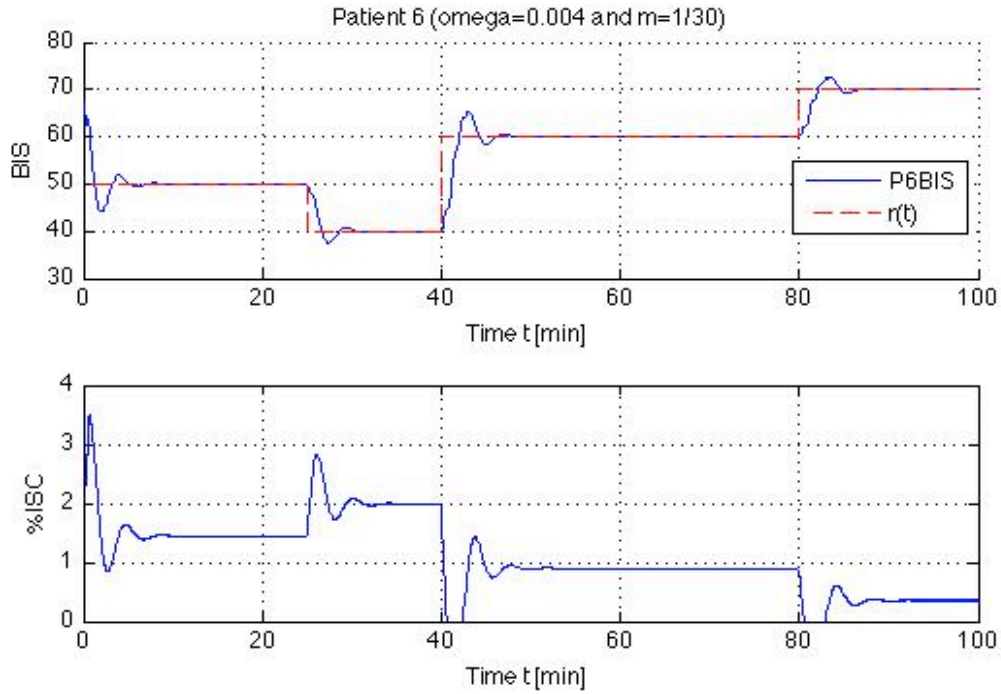
Figure 5.25: BIS reference tracking: Patient 6 output feedback controller with $\Gamma = 50000$

isoflurane use for Patient 6 is 3.285 liters.

Table 5.27: Patient 6 normalized residual errors: BIS tracking performance

ω/m	1/15	1/30	1/60
0.01	0.0023	0.0020	0.0020
0.008	0.0024	0.0021	0.0019
0.004	0.0027	0.0021	0.0020
0.002	0.0032	0.0027	0.0027
0.001	0.0039	0.0036	0.0035
0.0005	0.0048	0.0048	0.0048
0.0001	0.0116	0.0119	0.0122

5.2.5.2 Tracking, with disturbances

Just as before, the EP, EVAL, and LMA are treated as disturbance inputs to our ISO/BIS system. The transfer functions for the disturbances model of *Patient 6-sedated state* are:

74

Table 5.28: Patient 6 control effort: Isoflurane use in liters

ω/m	1/15	1/30	1/60
0.01	3.289	3.287	3.287
0.008	3.288	3.286	3.288
0.004	3.284	3.285	3.289
0.002	3.283	3.285	3.292
0.001	3.288	3.292	3.300
0.0005	3.303	3.307	3.315
0.0001	3.416	3.418	3.423

$$\frac{\text{BIS}}{\text{EP}} : \frac{-10.86s^3 - 7.739s^2 + 0.445s - 0.03246}{s^4 + 0.2513s^3 + 0.02248s^2 + 0.003408s + 2.944e - 05} \tag{5.47}$$

$$\frac{\text{BIS}}{\text{EVAL}} : \frac{-0.1031s^3 + 0.1144s^2 - 0.007212s + 0.001873}{s^4 + 0.2513s^3 + 0.02248s^2 + 0.003408s + 2.944e - 05} \tag{5.48}$$

$$\frac{\text{BIS}}{\text{LMA}} : \frac{1.098s^3 + 0.04537s^2 + 0.008963s + 0.00128}{s^4 + 0.2513s^3 + 0.02248s^2 + 0.003408s + 2.944e - 05} \tag{5.49}$$

As before, the disturbances effects on the BIS output are exactly the same as those of Patient 1. This was done for comparison and analysis purposes. Maintaining the same control design as discussed above, simulations were run with disturbance inputs. Figure 5.26 shows the closed-loop performance of the $\mathscr{L}_1$-adaptive controller used on Patient 6 when disturbances are introduced. This particular system results in a calculated normalized residual error with a value of 0.0032. Approximate isoflurane use for Patient 6 is 3.284 liters. For both simulations with and without disturbances, the system has a non-zero initial condition which leads to an exponentially decaying term in the control and system state signal; this does not affect the performance of the system throughout.

Tables 5.29 and 5.30 detail the performance attained by different controller configurations with disturbances to the system.

75

Table 5.29: Patient 6 with disturbances normalized residual errors: BIS tracking performance

ω/m	1/15	1/30	1/60
0.01	0.0032	0.0029	0.0027
0.008	0.0034	0.0029	0.0027
0.004	0.0040	0.0032	0.0027
0.002	0.0042	0.0037	0.0035
0.001	0.0051	0.0049	0.0043
0.0005	0.0065	0.0063	0.0059
0.0001	0.0122	0.0128	0.0130

Table 5.30: Patient 6 with disturbances control effort: Isoflurane use in liters

ω/m	1/15	1/30	1/60
0.01	3.288	3.285	3.286
0.008	3.287	3.285	3.287
0.004	3.285	3.284	3.289
0.002	3.283	3.285	3.293
0.001	3.289	3.292	3.300
0.0005	3.303	3.307	3.315
0.0001	3.416	3.419	3.423
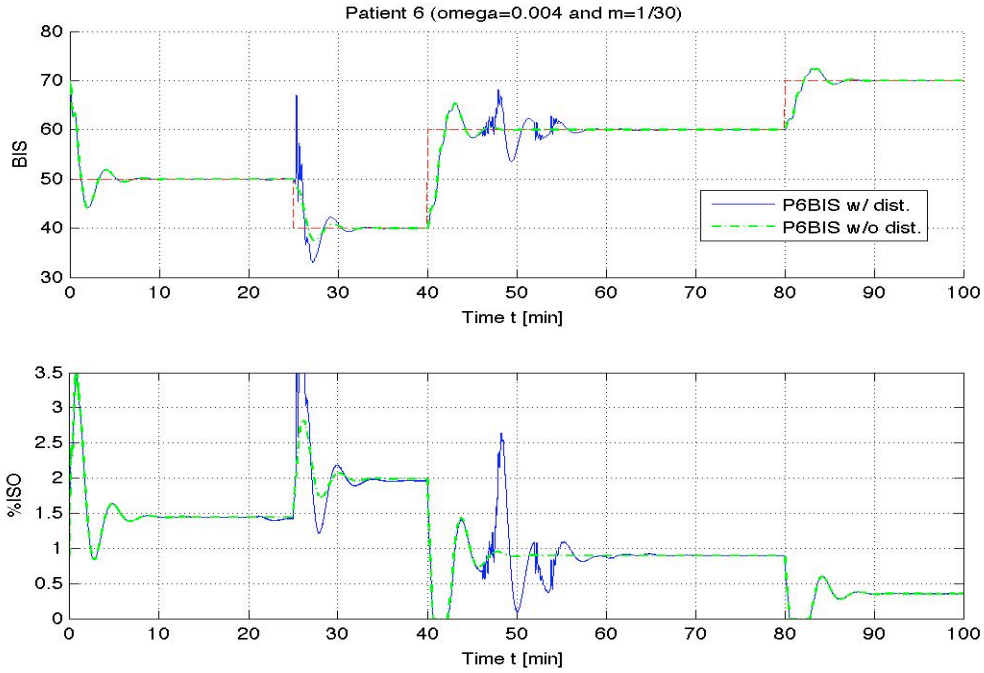
Figure 5.26: BIS reference tracking: Patient 6 output feedback controller with $\Gamma = 50000$ w/ disturbances

5.2.5.3 MAP performance, with disturbances

The MAP transfer functions for the model of *Patient 6-sedated state* are:

$$\text{ISO to BP: } \frac{2.006s - 0.008426}{s^2 + 0.1101s + 0.0008181} \tag{5.50}$$

$$\text{EVAL to BP: } \frac{-0.3818s - 0.06841}{s^2 + 0.1101s + 0.0008181} \tag{5.51}$$

$$\text{LMA to BP: } \frac{-2.94s + 0.1161}{s^2 + 0.1101s + 0.0008181} \tag{5.52}$$

77
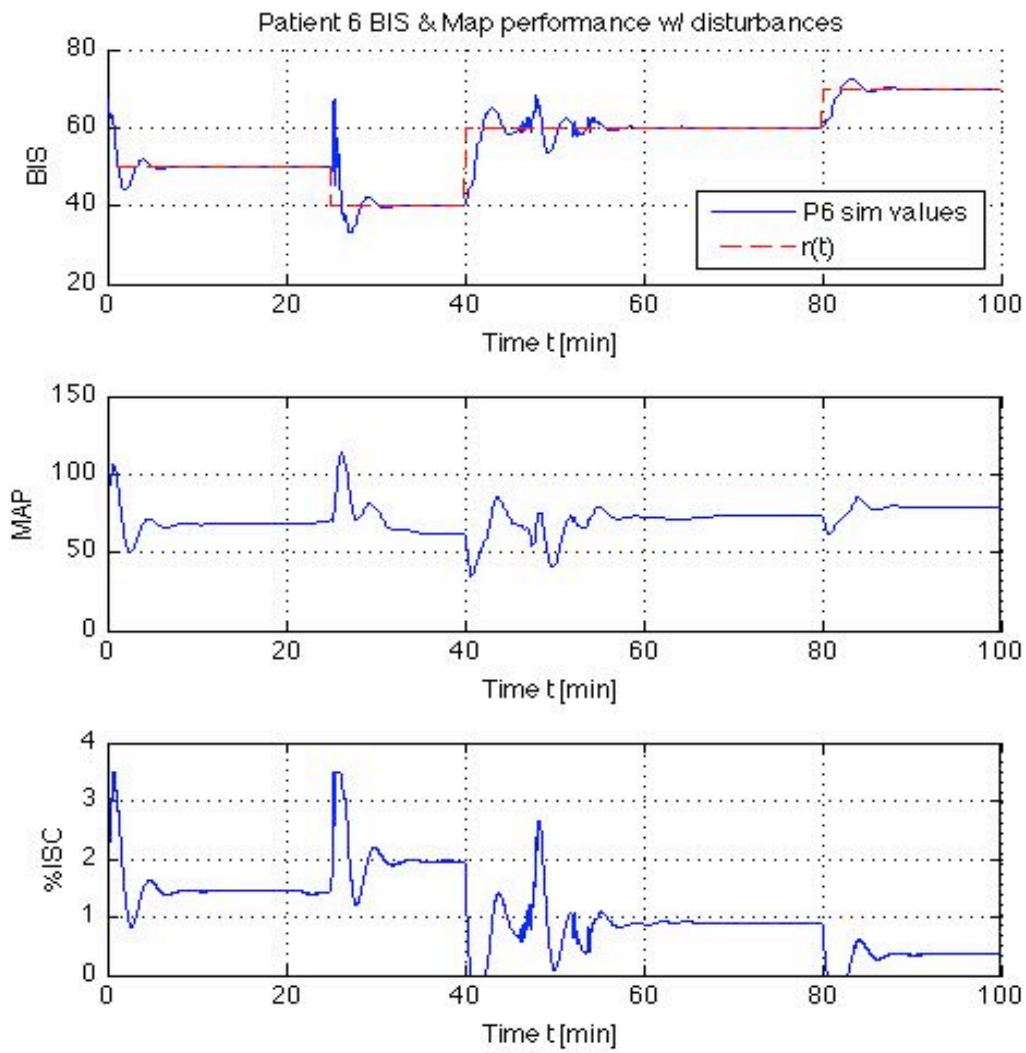
Figure 5.27: BIS reference tracking: Patient 6 BIS and MAP performance

Note, no EP event occurred when the BIS level of Patient 6 was below the threshold value of 70. As a result we do not have a transfer function for its effect on the BIS level. Figure 5.27 illustrates the BIS and MAP simulated trajectories when the controller designed for Patient 6 is applied. Here, the patient's MAP does drop below the minimum of the desired range a few times but never exceed the desired maximum bound.

5.2.5.4 Robustness to Inter-patient Variability

Similarly to our previous discussions, Patient 6's controller design was then simulated using the models for Patients 1, 2, 3, 5, and 7 to evaluate inter-patient design robustness. That is, we applied the controller designed for Patient 6 on Patients 1, 2, 3, 5, and 7. For Patients 2, 3, and 7, the patients' BIS levels tracked the desired reference profile closely, regardless of whether the applied controller was that specifically designed for them or not. When applied to Patients 1 and 5, the results were unsatisfactory. Upon closer examination, when m and ω are set as they are for the controller designed for Patient 6, $H(s)$ from (3.10) is not stable. Tables 5.31 and 5.32 detail the tracking performance and control effort when the controller designed for Patient 5 is applied to the models of Patients 1, 2, 3, 5, and 7. It is important to note that the controller is *exactly* the same for all patients.

Table 5.31: Normalized residual errors with Patient 6 control, no disturbances

Patient/Controller	6
1	N/A
2	0.0050
3	0.0020
5	N/A
7	0.0041

79

Table 5.32: Isoflurane use in liters with Patient 6 control, no disturbances

Patient/Controller	6
1	N/A
2	2.859
3	2.206
5	N/A
7	2.136

5.2.6   Patient 7 $\mathscr{L}_1$-Adaptive Control-Output Feedback

The *Patient 7-sedated* models were constructed through the MATLAB *n4sid* command on partitioned patient data. A MATLAB m-file was created to identify the values for all parameters required for the system in (2.4), and for the $\mathscr{L}_1$-adaptive controller given by (3.25), (3.26), and (3.29). Simulink was then used to simulate the closed-loop systems.

The model parameters for *Patient 7-sedated state* are:

$$
A = \begin{bmatrix} -0.0022 & -0.0307 & -0.0078 & -0.0123 \\ 0.0165 & -0.0759 & -0.0776 & -0.1197 \\ -0.0009 & 0.0170 & -0.0157 & -0.2734 \\ 0.0016 & 0.0253 & 0.2479 & 0.0159 \end{bmatrix},
$$

$$
B = \begin{bmatrix} -0.0016 \\ -0.0078 \\ -0.0118 \\ 0.0082 \end{bmatrix},
$$

$$
C = \begin{bmatrix} 175.3789 \\ -10.2101 \\ -0.8316 \\ 0.3883 \end{bmatrix}. \tag{5.53}
$$

5.2.6.1 Tracking, no disturbances

The transfer function for the model of *Patient 7-sedated state* are:

$$\frac{\text{BIS}}{\text{ISO}} : \frac{-0.188s^3 + 0.02198s^2 - 0.0101s - 0.0005796}{s^4 + 0.0779s^3 + 0.07254s^2 + 0.005276s + 4.678e - 05} \tag{5.54}$$

Again, it is assumed the ISO to BIS transfer function is strictly proper. We conservatively set $P = 1$ and chose $\Delta = 100$ and $\Gamma = 50000$. The $\mathscr{L}_1$-adaptive controller given by (3.25), (3.26), and (3.29) is applied to track a desired sedated BIS trajectory, $r(t)$. If for $M(s)$ in (2.7) we set $m = 1/60$ and $C(s)$ in (3.9) set $\omega = 0.004$, we can show that (3.10) is strictly proper and BIBO stable as required. Simulation results for the $\mathscr{L}_1$-adaptive controller applied to the dynamic response model for Patient 7 can be seen in Figure 5.28.
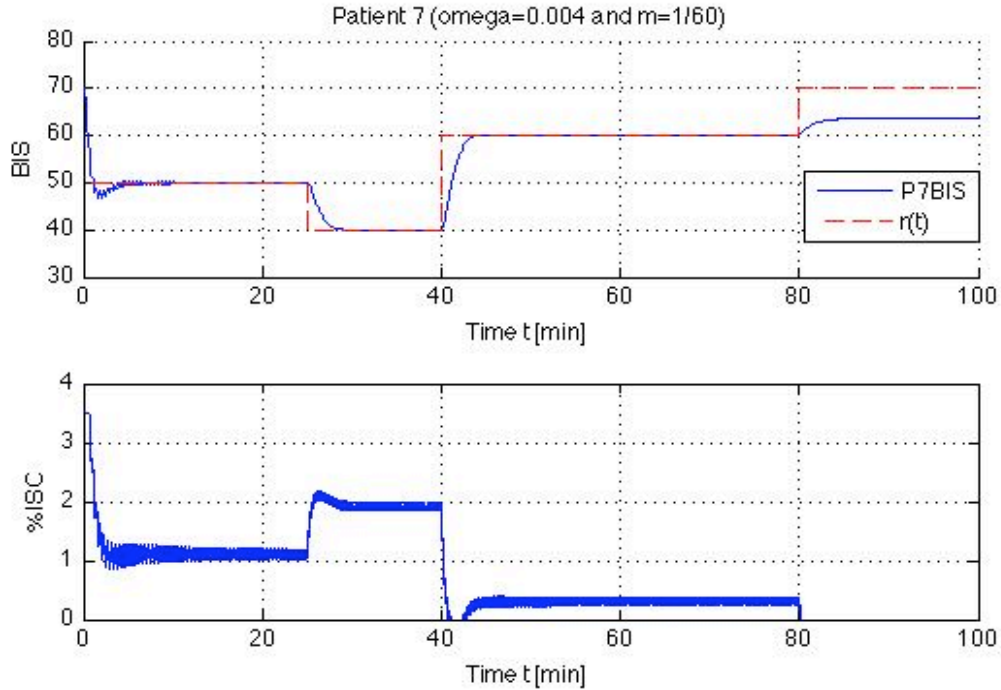
Figure 5.28: BIS reference tracking: Patient 7 output feedback controller with $\Gamma = 50000$

As can be seen in Figure 5.28, the $\mathscr{L}_1$-adaptive output feedback control design achieves the desired tracking performance quickly with little tracking error. This particular design achieves a normalized error residual of 0.0028. The approximate isoflurane use for Patient 7 is 2.133 liters. Tables 5.33 and 5.34 summarize results for different controller design configurations.

81

Table 5.33: Patient 7 normalized residual errors: BIS tracking performance

ω/m	1/15	1/30	1/60
0.02	0.0042	N/A	N/A
0.01	0.0043	0.0018	N/A
0.008	0.0044	0.0019	N/A
0.004	0.0048	0.0041	0.0028
0.002	0.0054	0.0055	0.0029
0.001	0.0065	0.0064	0.0056
0.0005	0.0079	0.0079	0.0072
0.0001	0.0134	0.0134	0.0133

Table 5.34: Patient 7 control effort: Isoflurane use in liters

ω/m	1/15	1/30	1/60
0.02	2.141	N/A	N/A
0.01	2.143	2.136	N/A
0.008	2.143	2.136	N/A
0.004	2.140	2.136	2.133
0.002	2.137	2.137	2.141
0.001	2.144	2.147	2.154
0.0005	2.170	2.173	2.180
0.0001	2.372	2.375	2.380

5.2.6.2 Tracking, with disturbances

Just as before, the EP, EVAL, and LMA are treated as disturbance inputs to our ISO/BIS system. The transfer functions for the disturbances model of *Patient 7-sedated state* are:

$$\frac{\text{BIS}}{\text{EP}} : \frac{-10.65s^3 - 6.611s^2 - 0.01062s - 0.3286}{s^4 + 0.0779s^3 + 0.07254s^2 + 0.005276s + 4.678e - 05} \tag{5.55}$$

$$\frac{\text{BIS}}{\text{EVAL}} : \frac{-1.497s^3 + 0.04269s^2 - 0.1074s + 0.002277}{s^4 + 0.0779s^3 + 0.07254s^2 + 0.005276s + 4.678e - 05} \tag{5.56}$$

$$\frac{\text{BIS}}{\text{LMA}} : \frac{0.5809s^3 - 0.1163s^2 + 0.04041s + -0.005745}{s^4 + 0.0779s^3 + 0.07254s^2 + 0.005276s + 4.678e - 05} \tag{5.57}$$

As before, the disturbances effects on the BIS output are exactly the same as those of Patient 1. This was done for comparison and analysis purposes. Maintaining the same control design as previously discussed, simulations were run with disturbance inputs. Figure 5.29 shows the closed-loop performance of the $\mathscr{L}_1$-adaptive controller used on Patient 7 when disturbances are introduced. This particular system results in a calculated normalized residual error with a value of 0.0030. Approximate isoflurane use for Patient 7 is 2.135 liters. For both simulations with and without disturbances, the system has a non-zero initial condition which leads to an exponentially decaying term in the control and system state signal; this does not affect the performance of the system throughout.

83
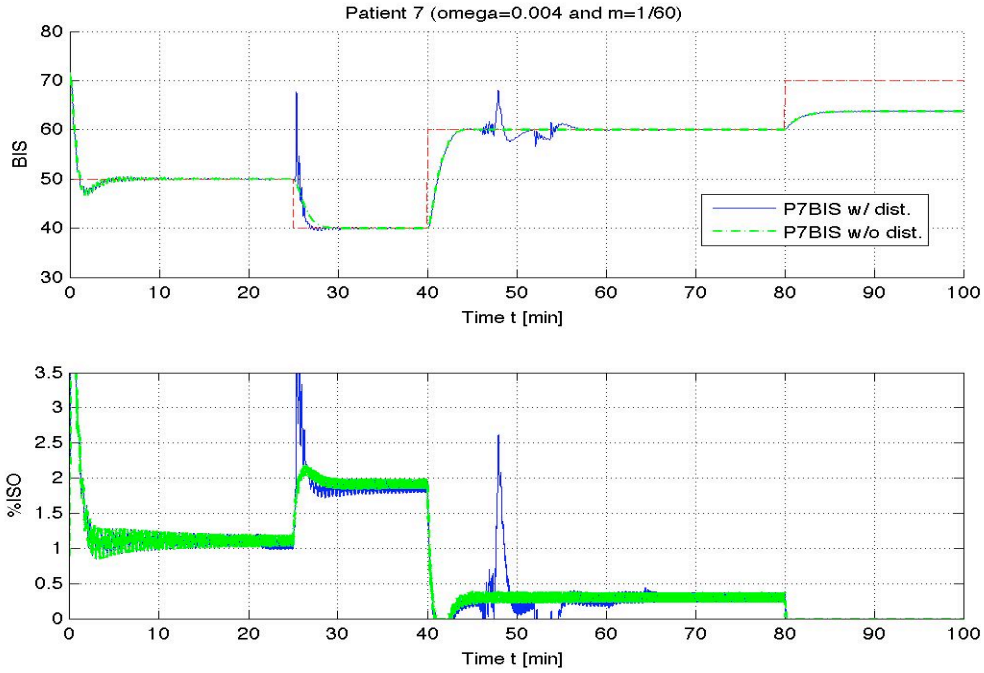
Figure 5.29: BIS reference tracking: Patient 7 output feedback controller with $\Gamma = 50000$ w/ disturbances

In Tables 5.35 and 5.36 we have results of various design configurations for BIS level tracking with disturbances to the patient.

Table 5.35: Patient 7 with disturbances normalized residual errors: BIS tracking performance

ω/m	1/15	1/30	1/60
0.02	0.0041	N/A	N/A
0.01	0.0048	0.0022	N/A
0.008	0.0049	0.0025	N/A
0.004	0.0054	0.0049	0.0030
0.002	0.0064	0.0065	0.0037
0.001	0.0074	0.0075	0.0062
0.0005	0.0088	0.0088	0.0079
0.0001	0.0139	0.0142	0.0139

Table 5.36: Patient 7 with disturbances control effort: Isoflurane use in liters

ω/m	1/15	1/30	1/60
0.02	2.141	N/A	N/A
0.01	2.145	2.138	N/A
0.008	2.146	2.139	N/A
0.004	2.147	2.140	2.135
0.002	2.142	2.140	2.142
0.001	2.146	2.148	2.154
0.0005	2.171	2.174	2.180
0.0001	2.374	2.377	2.381

5.2.6.3 MAP performance, with disturbances

Mean arterial pressure measurements do not have to track a specific reference trajectory. Instead, during surgery, there is a range the anesthesiologist would like the patient's MAP to remain in. In our study, we would like the MAP to stay in the 60 to 110 mmHg range. The MAP transfer functions for the model of *Patient 7-sedated state* are:

$$\text{ISO to MAP: } \frac{0.0805s^2 + 0.01302s + 2.383e - 05}{s^3 + 0.1454s^2 + 0.0009549s + 5.43e - 07} \tag{5.58}$$

$$\text{EP to MAP: } \frac{-5.168e - 14s^2 - 7.208e - 15s}{s^3 + 0.1454s^2 + 0.0009549s + 5.43e - 07} \tag{5.59}$$

$$\text{EVAL to MAP: } \frac{-0.0872s^2 - 0.01237s3.338e - 05}{s^3 + 0.1454s^2 + 0.0009549s + 5.43e - 07} \tag{5.60}$$

$$\text{LMA to MAP: } \frac{-1.521s^2 - 0.03287s - 0.0004821}{s^3 + 0.1454s^2 + 0.0009549s + 5.43e - 07} \tag{5.61}$$

Figure 5.30 illustrates the BIS and MAP performance achieved when the $\mathscr{L}_1$-adaptive output feedback controller with the previously detailed parameters is applied to Patient 7. Clearly, Patient 7's MAP is in our desired range for the first 60 minutes of the simulation, after which it drops below the desired minimum and settles at approximately 45 mmHg. This is the only patient out of our set of 6 that exhibits this out of range behavior.

86
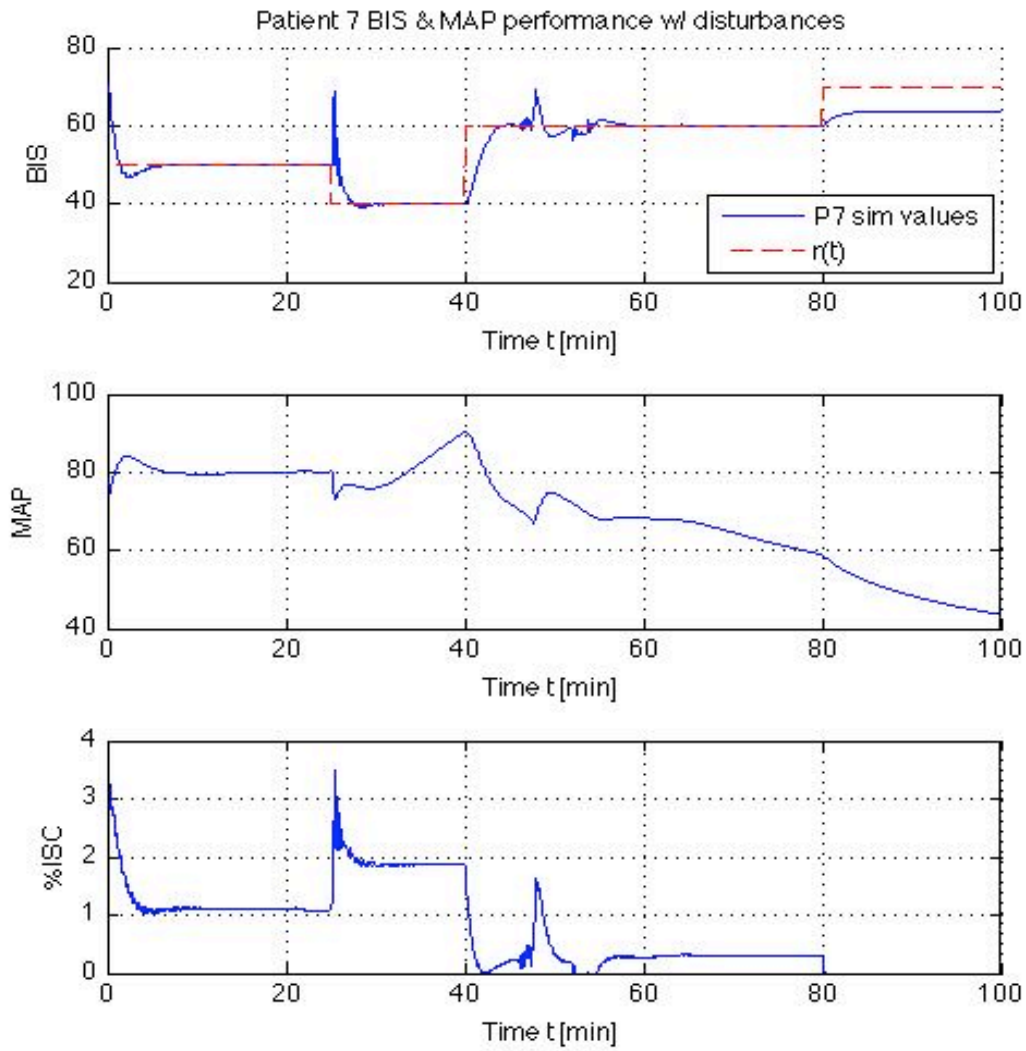
Figure 5.30: BIS reference tracking: Patient 7 BIS and MAP performance

5.2.6.4 Robustness to Inter-patient Variability

As before, Patient 7's controller design was then simulated using the models for Patients 1, 2, 3, 5, and 6 to evaluate inter-patient design robustness. That is, we applied the controller designed for Patient 7 on Patients 1, 2, 3, 5, and 6. For Patients 2, 3, and 7, the patients' BIS levels tracked the desired reference profile closely, regardless of whether the applied controller was that specifically designed for them or not. When applied to Patients 1 and 5, the results were unsatisfactory. Upon closer examination, when m and ω are set as they are for the controller designed for Patient 7, $H(s)$ from (3.10) is not stable. Tables 5.37 and 5.38 detail the tracking performance and control effort when the controller designed for Patient 7 is applied to the models of Patients 1, 2, 3, 5, and 6. It is important to note that the controller is *exactly* the same for all patients.

Table 5.37: Normalized residual errors with Patient 7 control, no disturbances

Patient/Controller	7
1	N/A
2	0.0050
3	0.0020
5	N/A
6	0.0041

Table 5.38: Isoflurane use in liters with Patient 7 control, no disturbances

Patient/Controller	7
1	N/A
2	2.859
3	2.206
5	N/A
6	2.136

5.2.7 Effect of the order of filter $C(s)$ on system performance

In this section we examine the effects that increasing the order of filter $C(s)$ have on the performance of the $\mathscr{L}_1$-adaptive output feedback scheme. When possible,

88

it is desired that the oscillations in the control channel be minimized as this would make implementation easier. We evaluated the extent of such oscillations and tracking performance with first order and second order filter designs on a subset of the patients. We present these preliminary findings below.

With the first-order filters, the control signals of Patients 1, 3, and 7 oscillate more than is desired. While these signals do not oscillate at a high frequency, if the oscillations in the control channel can be reduced, that is preferred.

An example of the closed-loop system behavior when simulating Patient 1's model with a first-order filter is shown in Figure 5.9. While the tracking performance is good, the input signal ideally should be cleaner. The input signal does not oscillate at a high frequency, but we would like to further minimize the extent of oscillation as much as possible, as this would make real-world application easier; depending on hardware limitations, the device that delivers the anesthesia to the patient may not be able to generate the same anesthesia profile as the control signal if the signal oscillates at too large of a frequency. To address this problem, a higher order filter $C(s)$ is designed and applied to the same output predictor, adaptive law, and control law as discussed in the previous sections. Simulating the system with the filter given in equation (5.62) yields an r_N value of 0.0032 and results in 2.407 liters of isoflurane consumed.

$$C(s) = \frac{0.0064}{s^2 + 10s + 0.0064} \tag{5.62}$$

This second-order filter attains the tracking performance shown in figure 5.31. The tracking performance is very similar to that obtained with the first-order system, shown in figure 5.9, albeit with a cleaner input signal.

Another second-order Patient 1 filter design is

$$C(s) = \frac{0.01}{s^2 + 10s + 0.01}. \tag{5.63}$$

The attained normalized error residual r_N with the filter given in (5.63) is 0.0031. The total amount of ISO consumed is 2.404 liters. Note, the second-order filters $C(s)$ given in (5.62) and (5.63) were designed to have a similar frequency response to the first-order filter previously discussed in the Patient 1 $\mathscr{L}_1$-adaptive output feedback controller design.

The Patient 3 system exhibits the most oscilliatory input signal of our sample of 6 patients when the output-feedback scheme is applied. As with the Patient 1 model, we designed and implemented a second-order filter to determine possible
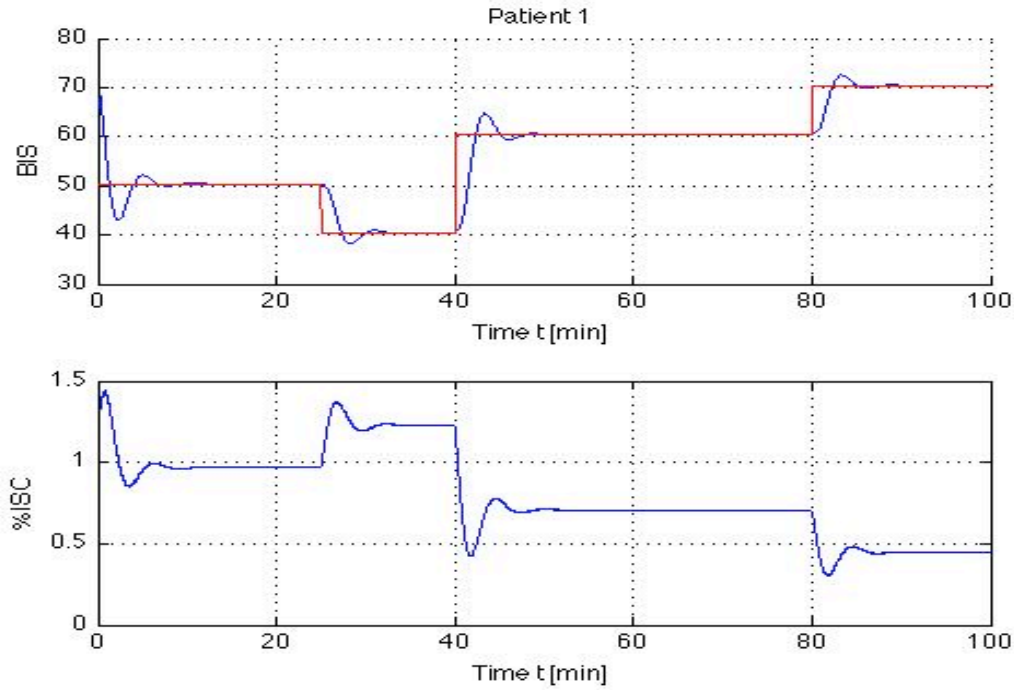
Figure 5.31: BIS reference tracking: Patient 1 output feedback controller with second-order filter (5.62)

improvement on the input signal. The filter given in equation (5.64) is designed to give similar tracking performance as the first-order filter previously discussed for the Patient 3 $\mathscr{L}_1$-adaptive output feedback controller design; this filter is applied and simulated. Figure 5.33 illustrates the effects that the second order filter given in equation (5.64) has on the Patient 3 system.

$$C(s) = \frac{0.0064}{s^2 + 0.288s + 0.0064}, \tag{5.64}$$

The attained error residual r_N with filter (5.64) is $4.4948 * 10^{-4}$. The total amount of ISO consumed to follow the reference BIS trajectory is 2.302 liters.

Comparing these results with the performance of the system when using the first order filter,

$$C(s) = \frac{0.008}{s + .008}. \tag{5.65}$$

for which 2.305 liters of isoflurane is consumed and an r_N of $5.2742 * 10^{-4}$, it can be seen that using the second-order filter achieves similar performance with approximately the same isoflurane consumption. However, the system input signal has significantly less oscillation than with the first-order filter, with comparable
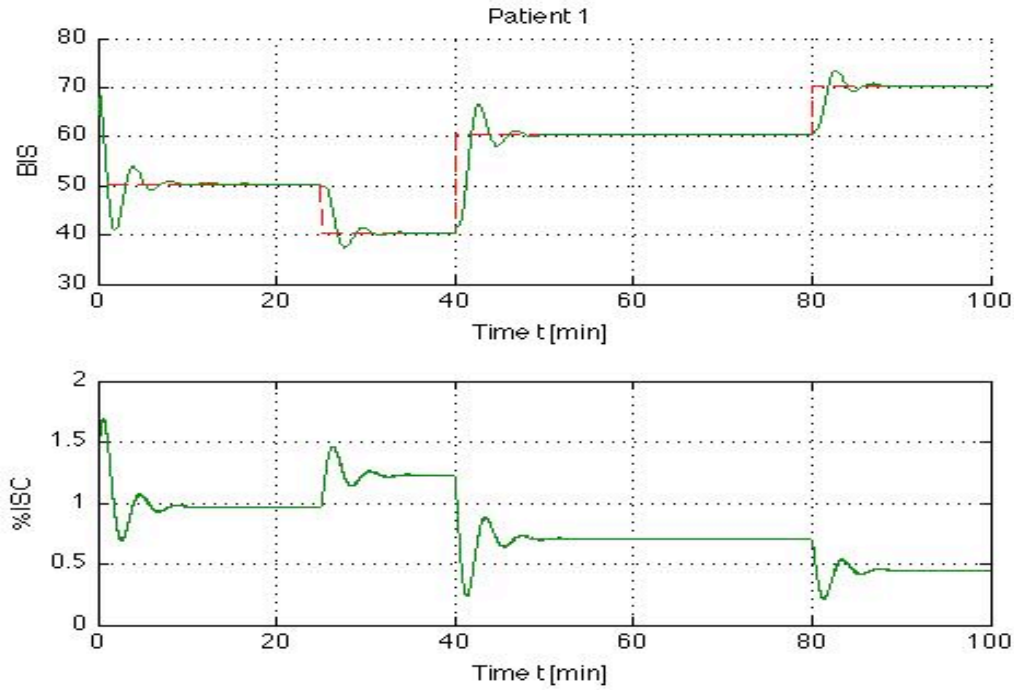
Figure 5.32: BIS reference tracking: Patient 1 output feedback controller with second-order filter (5.63)

performance. This fact is obviated by comparing Figures 5.33 and 5.17.

The Patient 7 system also exhibits an input signal that has more oscillatory behavior than we desire. As with the Patients 1 and 3 models, we designed and implemented a second-order filter to determine possible improvement on the input signal. The filter given in equation (5.66) is designed to give similar step-response performance as the first-order filter previously discussed for the Patient 7 $\mathscr{L}_1$-adaptive output feedback controller design. this filter is applied and simulated. Figure 5.34 demonstrates the effects that the second-order filter given in equation (5.66) has on the Patient 7 system.

$$C(s) = \frac{0.01}{s^2 + 3s + 0.01}, \tag{5.66}$$

91
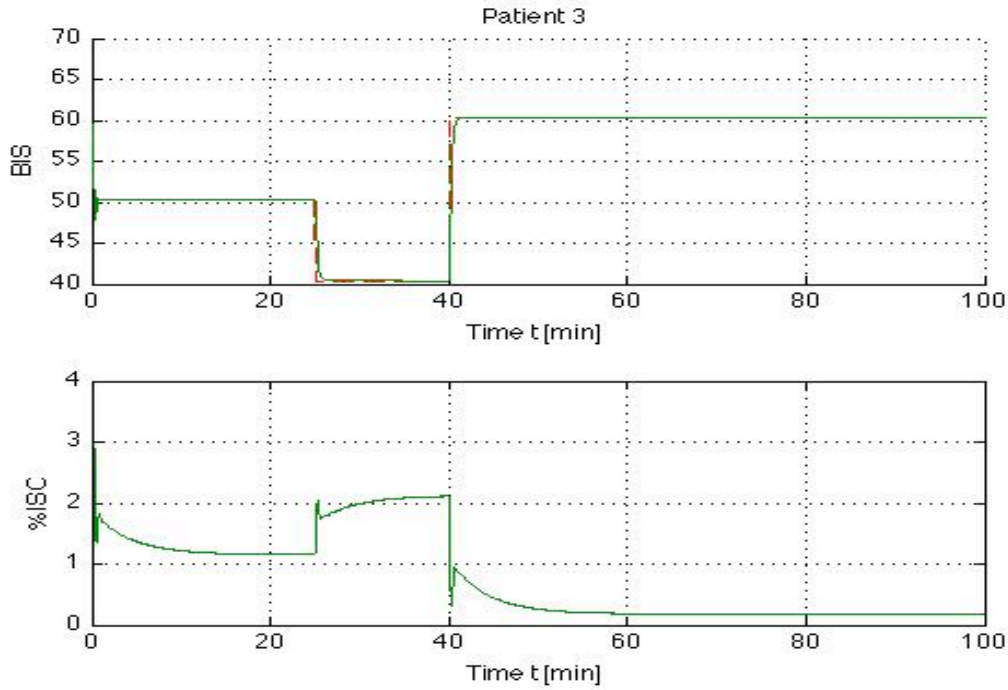
Figure 5.33: BIS reference tracking: Patient 3 output feedback controller with second-order filter (5.64)

The attained error residual r_N with filter (5.66) is 0.0044. The total amount of ISO consumed to follow the reference BIS trajectory is 2.135 liters.

Comparing these results with the performance of the system when using the first order filter,

$$C(s) = \frac{0.004}{s + .004}.$$ (5.67)

for which 2.133 liters of isoflurane is consumed and an r_N of 0.0028, it can be seen that using the second-order filter achieves similar performance with approximately the same isoflurane consumption. However, the system input signal has significantly less oscillation than with the first-order filter, with comparable performance. This fact is obviated by comparing Figures 5.34 and 5.28.

With second-order filters, it appears we can attain nearly identical tracking performance (and depending on the patient model and how the filter is designed, improved performance) and improved control effort performance. The input signals are much cleaner in that they have reduced oscillation than the first-order filters which yield similar tracking performance. This is particularly useful since this makes the controller more feasible and easier to implement in real-world environments.

92
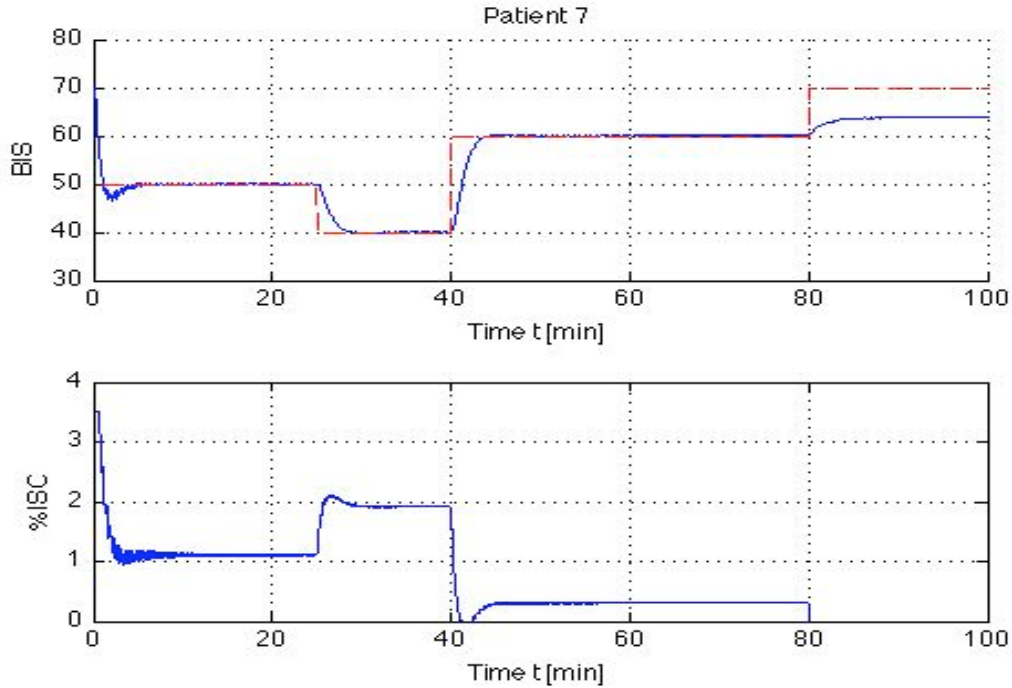
Figure 5.34: BIS reference tracking: Patient 7 output feedback controller with second-order filter (5.66)

5.3 LPV vs. $\mathscr{L}_1$-adaptive control performance

An LPV control approach presented and discussed in [42]. In this subsection we compare the resulting performance results.

In terms of reference tracking performance, the $\mathscr{L}_1$-adaptive controllers exhibit better tracking performance than the LPV controllers. Comparing the tracking results, for example as shown in Figures 5.35, 5.36, and 5.37, it appears that the $\mathscr{L}_1$-adaptive control schemes for the most part attain and follow the desired BIS reference, but the LPV controllers only maintain the output signals near the desired BIS levels. The normalized error residuals for all three control scheme can be seen in Table 5.39.

In Table 5.40, the estimated control effort required for each of the three control schemes are given. The $\mathscr{L}_1$-adaptive schemes require less anesthesia in call cases except Patient 6. However, note that in the LPV study, Patient 6's BIS never drops below 60. In both $\mathscr{L}_1$-adaptive simulations, the patient's BIS spends a significant portion of the trial time at or below a BIS level of 50.

Also of interest is a comparison of the two control approaches in robustness to interpatient variability. The results given in [42] indicate that the LPV controller

93

Table 5.39: BIS Reference Tracking Comparison

Patient	LPV	L1-SF	L1-OF
1	0.1119	0.0026	0.0019
2	0.2161	0.0021	0.0050
3	0.1095	0.0044	0.0015
5	0.1194	0.0032	0.0012
6	0.1086	0.0029	0.0021
7	0.1309	0.0033	0.0028

Table 5.40: Control Effort Comparison

Patient	LPV	L1-SF	L1-OF
1	2.745	2.391	2.404
2	4.425	2.035	2.853
3	3.607	2.754	2.197
5	2.850	2.338	2.397
6	2.820	3.248	3.285
7	2.820	2.793	2.133
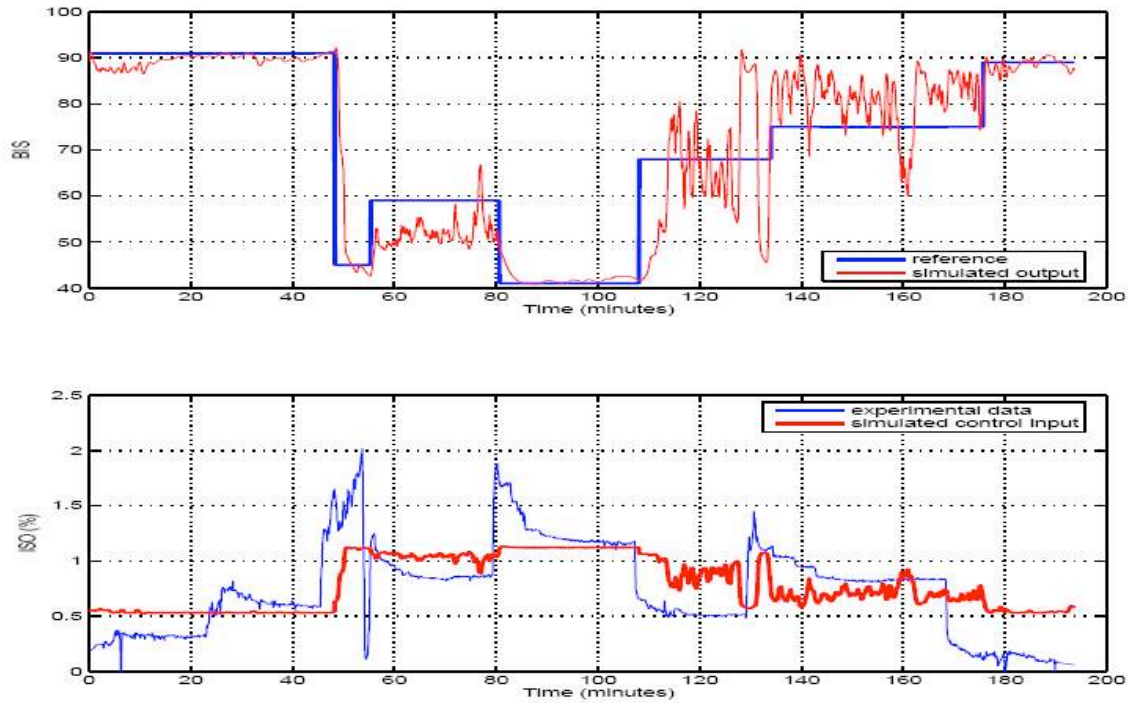
Figure 5.35: LPV control simulation result of BIS reference tracking for Patient 1

generated for Patient 1 can be used as a controller for Patient 2 and 6, and the LPV controller generated for Patient 3 can be used as a controller for Patients 5 and 7. The $\mathscr{L}_1$-adaptive state-feedback control scheme appears to yield more robust controllers. As can be seen in Tables 5.1 and 5.2, the $\mathscr{L}_1$-adaptive scheme, for example, allows for the controller designed for Patient 1 used on Patients 2, 5, 6 and 7 with no deterioration in results. Similarly, the $\mathscr{L}_1$-adaptive controller designed for Patient 3 can be applied to Patients 1, 2, 5, 6, and 7 and results in good tracking performance. The same follows for five of the six controllers designed for our patient set. The $\mathscr{L}_1$-adaptive output-feedback control scheme appears to be the have the greatest robustness to interpatient variability; the $\mathscr{L}_1$-adaptive output-feedback control scheme yields controllers that can be applied to the entire patient set, and with no deterioration to the tracking performance. That is, you could take any $\mathscr{L}_1$-adaptive output-feedback controller designed for any of the patients in our set, apply it any other patient in the set, and achieve the desired BIS tracking performance.

It is important to note that there are two different sets of assumptions in the LPV and $\mathscr{L}_1$-adaptive control schemes. The LPV controllers are designed to control the patient during both the awake and sedated states. The $\mathscr{L}_1$-adaptive control

95
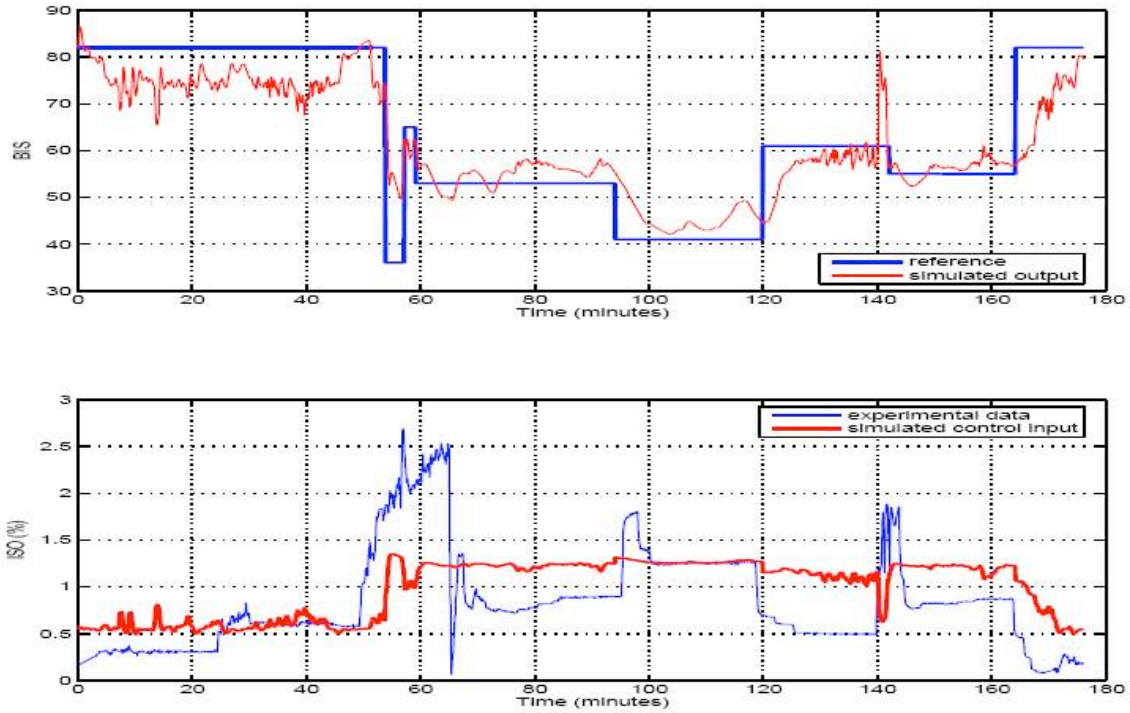
Figure 5.36: LPV control simulation result of BIS reference tracking for Patient 3

scheme however assumes that the anesthesiologist first brings the patient into the sedated region and then the controller is switched on. In the sedated region, which is where the control is most required during the surgery, the $\mathscr{L}_1$-adaptive control schemes provide better tracking performance, use less anesthesia, and appear to be more robust to interpatient variability (particularly in the case of the $\mathscr{L}_1$-adaptive output-feedback scheme).
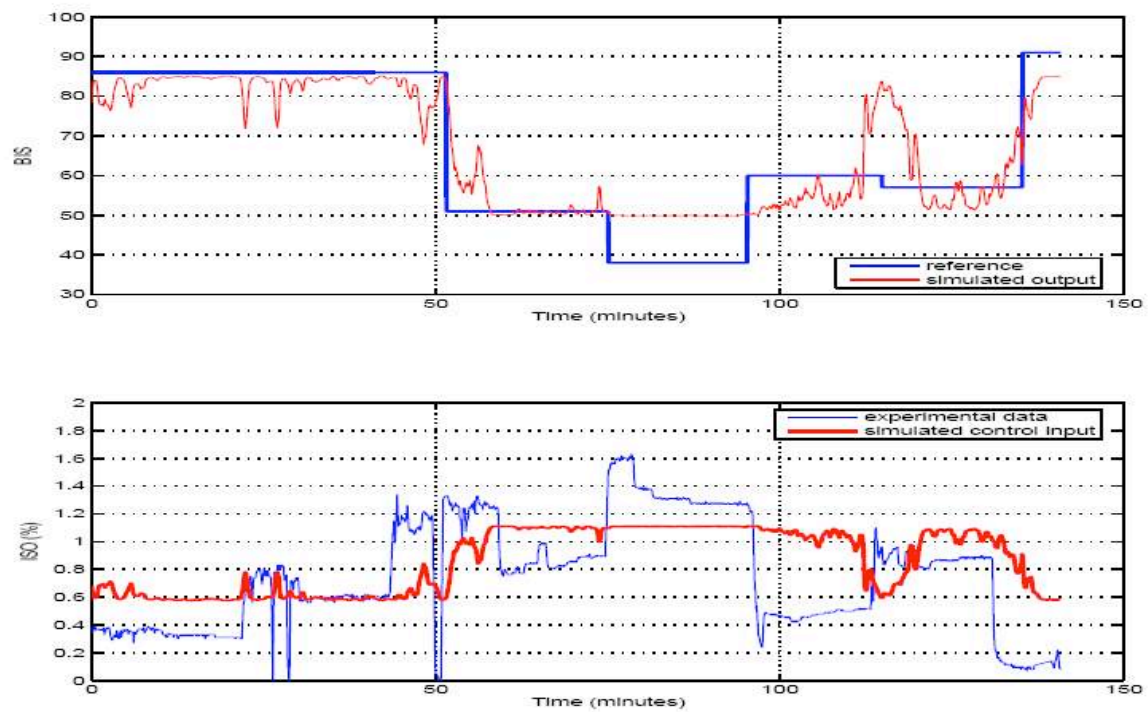
Figure 5.37: LPV control simulation result of BIS reference tracking for Patient 5

CHAPTER 6

CONCLUSIONS AND ONGOING RESEARCH

In this thesis, we present the first application of recently proposed $\mathscr{L}_1$-adaptive control techniques to the anesthesia control problem. The acheived tracking performance and inter-patient robustness of these control methods, in conjunction with the relatively effective use of isoflurane, is extremely encouraging, and MIMO design efforts incorporating surgical stimuli inputs and additional performance objectives on patient vital signs are ongoing.

6.1 State vs. Output Feedback

One important aspect to consider when comparing the state-feedback $\mathscr{L}_1$-adaptive scheme versus the output-feedback in our studies is the meaning of the values being fed back to the controller. In the state-space scheme, the models we identified were black-box models. The states do not have any direct physical meaning attached to them. So the question becomes how would one use this outside of theory and actually apply it? The output feedback controller described in section 3.3 requires only the BIS value be known. Since the BIS is a measureable, physical signal, this is far more useful in terms of building and implementing the controller in real-world applications.

Robustness to interpatient variability is also important to examine. When designing a control system to administer anesthesia to surgery patients, a design that is as robust as possible is desired, as the difference in response to anesthesia and surgical stimuli can vary greatly from one person to the next. Based on the the simulations conducted for this paper, the output feedback scheme appears betters suited for this task than the state feedback scheme. The state-feedback scheme requires the desired reference model to have certain structural characteristics in common with those of the plant, thus it is more restrictive in applicability to all of the patient models. The output feedback scheme, on the other hand, controls

the system input based on the *behavior* of the system in that controllers are designed based on transfer functions and not specific matrices like the state-feedback scheme does.

Performance when disturbances to the system are present is also an important factor to consider. For example, consider a common surgery scenario. The patient is subjected to a variety of stimuli that could cause the patient to wake up. These include, but are not limited to, incisions, jostling of the patient, intubation, noises that occur during surgery, and suturing. It is essential that while the patient is undergoing surgery they remain sedated. This is not only physically safer for the patient, but also prevents emotional or psychological trauma that could result from waking up during such an event. Thus there is an important question that must be answered: which control scheme best addresses this issue? Based on the simulations performed and the tracking and control effort data gathered from these simulations, the output feedback scheme performs better than the state feedback. As previously mentioned, the output feedback relies strictly on the BIS signal to determine the amount of anesthesia to administer. As such, when the patients BIS starts to rise due to some disturbance, the controller will administer more anesthesia to counter this effect. The state-feedback scheme however relies on the state to determine the anesthesia level. The problem lies in the fact that the structure of the models results in very little state change when the disturbances enter the system. This is a problem because the state feedback relies on minimizing the *state* error rather than the output error.

In terms of tracking performance (i.e. settling time, peak overshoot, steady-state error, etc.), the state-feedback scheme appears to more easily obtain better results than the output-feedback scheme. If the tables provided in Chapter 5 are examined, the r_N values illustrate this. It is, however, important to note the state-feedback scheme's robustness to interpatient variability and disturbances inherent in a surgery make it considerably less suitable for our purposes than the output-feedback. The output-feedback scheme itself achieves very good tracking, both with and without disturbances.

Based on the previously discussed characteristics, the output-feedback scheme appears to be the best suited of the two architectures for the anesthesia problem. The $\mathscr{L}_1$-adaptive output feedback control scheme's tracking performance, ability to be applied to different patients with no adjustments when the reference system $M(s)$ and filter $C(s)$ is conservatively tuned, ability to counteract disturbances to the BIS profile, and reliance on a measureable physical signal for feedback make

$\mathscr{L}_1$-adaptive output feedback scheme a more feasible and overall useful approach to this control problem.

6.2 Observations

When investigating robustness of output feedback control designs to interpatient variability, it appears that to guarantee applicability of a single controller to more than just one patient, one must conservatively design $C(s)$ and $M(s)$. As usual, there is a trade-off between making the controller robust and having high performance (in our case, this generally results in a decrease in the ability of the controller to cancel the effects of disturbances on the BIS value). Conversely, with a more conservative controller, it appears more likely that vital signs like heart-rate and mean arterial pressure will remain in the desired range of values. For controllers with stronger tracking performance, the control action is more aggressive. That is, the controller administers larger amounts of anesthesia over short periods of time when responding to differences between $r(t)$ and the patient's BIS value. As can be seen in many of the tables from the previous chapter, there are a number of combinations of $C(s)$ and $M(s)$ that can be used in the controller for Patients 1, 2, 3, 5, 6, and 7. Patients 1 and 5, however, require a more conservative $C(s)$ filter than the others. In most patients' control designs we can use a filter like (3.9) with $\omega = 0.002$ and generally any of the desired reference models of the form (2.7) with the m values we have investigated in this thesis.

Based on the the preliminary study into higher order filters in Section 5.2.7, it also appears that increasing the order of the filter, $C(s)$, can lead to improved closed-loop performance using the $\mathscr{L}_1$-adaptive output-feedback controller. Higher order filters can improve the quality of the input signal by reducing oscillations in the control channel. Higher order filters can also improve on the tracking performance over first order filters. This improvement in control effort and tracking performance does come at a price, however, as increasing the order of the filter increases the complexity of the system and the difficulty of designing the controller, particularly when evaluating the stability conditions. Increasing the order of the $C(s)$ filter from first to second order, however can improve performance with a relatively small increase in complexity.

100

6.3 Future Work

Higher order filters in the $\mathscr{L}_1$ design process should be explored further to determine to what extent they improve performance. The emphasis of this study was to determine whether or not the $\mathscr{L}_1$ approach to control design delivered robustness in terms of a single control design being applicable to many different patients. While our results demonstrate how robust this control architecture can be, it remains to be investigated what yields the best performance for the greatest number of patients. Surgical data will be used to test this control approach under harsher conditions, (i.e. greater number of anesthetic agents, disturbances, and actual surgical stimuli events).

APPENDIX A

SYSTEM IDENTIFICATION MATLAB CODE

A.1   n4sid

Patient 1 BIS

Load patient 1 data

```
load patient1-2ipt-v3.mat
```

Create data

```
z = iddata
(used_bis_data', [used_iso_data' used_EP_data' used_EVAL_data'used_LMA_data'],5);
z.inputn={ 'ISO';'EP';'EVAL';'LMA' };
z.outputn={ 'BIS' };
```

Partition data

```
zawake=[z(1:732,:); z(1674:2563,:)];
zsleep=z(733:1674,:);
zawakev=zawake(1:732);
zawakee=zawake(733:1622);
zsleepe=zsleep(1:471);
zsleepv=zsleep(472:942);
```

Detrend

```
zawakee=detrend(zawakee);
zawakev=detrend(zawakev);
zsleepe=detrend(zsleepe);
zsleepv=detrend(zsleepv);
```

Find Order of Sleep

```
msleep=n4sid(zsleepe,1:15);
```

Find Order of Awake

```
mawake=n4sid(zawakee,1:15);
```

Impulse Response of Sleep Model

impulse(msleep,'sd',3,'fill');

Impulse Response of Awake Model

impulse(mawake,'sd',3,'fill');

Find Awake Model using n4sid 4 states

mawake4=n4sid(zawakee,4,'nk',[1 0 0 0], 'Focus','Simulation');

[mawake4B,Da4] = BALREAL(mawake4);

Find Sleep Model using n4sid 4 states

msleep4=n4sid(zsleepe,4,'nk',[1 0 0 0], 'Focus','Simulation');

[msleep4B,Ds4] = BALREAL(msleep4);

Find Awake Model using n4sid 5 states

mawake5=n4sid(zawakee,5,'nk',[1 0 0 0], 'Focus','Simulation');

[mawake5B,Da5] = BALREAL(mawake5);

Find Sleep Model using n4sid 5 states

msleep5=n4sid(zsleepe,5,'nk',[1 0 0 0], 'Focus','Simulation');

[msleep5B,Ds5] = BALREAL(msleep5);

Find Awake Model using n4sid 6 states

mawake6=n4sid(zawakee,6,'nk',[1 1 0 0], 'DisturbanceModel','None');

[mawake6B,Da6] = BALREAL(mawake6);

Find Sleep Model using n4sid 6 states

msleep6=n4sid(zsleepe,6,'nk',[1 1 0 0], 'DisturbanceModel','None');

[msleep6B,Ds6] = BALREAL(msleep6);

Find Awake Model using n4sid 7 states

mawake7=n4sid(zawakee,7,'nk',[1 1 0 0], 'DisturbanceModel','None');

[mawake7B,Da7] = BALREAL(mawake7);

Find Sleep Model using n4sid 7 states

msleep7=n4sid(zsleepe,7,'nk',[1 1 0 0], 'DisturbanceModel','None');

[msleep7B,Ds7] = BALREAL(msleep7);

Compare models with zsleepv and zawakev

Compare mawake4 with zawakev

compare(zawakev,mawake4);

Compare mawake5 with zawakev

compare(zawakev,mawake5);

Compare mawake6 with zawakev

compare(zawakev,mawake6);

Compare mawake7 with zawakev

103

```
compare(zawakev,mawake7);
```
Compare msleep4 with zsleepv
```
compare(zsleepv,msleep4);
```
Compare msleep5 with zsleepv
```
compare(zsleepv,msleep5);
```
Compare msleep6 with zsleepv
```
compare(zsleepv,msleep6);
```
Compare msleep7 with zsleepv
```
compare(zsleepv,msleep7);
```

Patient 1 BP

Load patient 1 data
```
load patient1-2ipt-v3.mat
```
Create data
```
z=iddata
(used_bp_data',[used_iso_data' used_EP_data' used_EVAL_data' used_LMA_data'],5);
z.inputn={'ISO';'EP';'EVAL';'LMA'};
z.outputn={'BP'};
```
Partition data
```
zawake=[z(1:732,:); z(1674:2563,:)];
zsleep=z(733:1674,:);
zawakev=zawake(1:732);
zawakee=zawake(733:1622);
zsleepe=zsleep(1:471);
zsleepv=zsleep(472:942);
```
Detrend
```
zawakee=detrend(zawakee);
zawakev=detrend(zawakev);
zsleepe=detrend(zsleepe);
zsleepv=detrend(zsleepv);
```
Find Order of Sleep
```
msleep=n4sid(zsleepe,1:15);
```
Find Order of Awake
```
mawake=n4sid(zawakee,1:15);
```

104

Impulse Response of Sleep Model
impulse(msleep,'sd',3,'fill');
Impulse Response of Awake Model
impulse(mawake,'sd',3,'fill');
Find Awake Model using n4sid 1 states
mawake4=n4sid(zawakee,1,'nk',[1 0 0 0], 'Focus','Simulation');
[mawake4B,Da4] = BALREAL(mawake4);
Find Sleep Model using n4sid 1 states
msleep4=n4sid(zsleepe,1,'nk',[1 0 0 0], 'Focus','Simulation');
[msleep4B,Ds4] = BALREAL(msleep4);
Find Awake Model using n4sid 2 states
mawake5=n4sid(zawakee,2,'nk',[1 0 0 0], 'Focus','Simulation');
[mawake5B,Da5] = BALREAL(mawake5);
Find Sleep Model using n4sid 2 states
msleep5=n4sid(zsleepe,2,'nk',[1 0 0 0], 'Focus','Simulation');
[msleep5B,Ds5] = BALREAL(msleep5);
Find Awake Model using n4sid 6 states
mawake6=n4sid(zawakee,6,'nk',[1 1 0 0], 'DisturbanceModel','None');
[mawake6B,Da6] = BALREAL(mawake6);
Find Sleep Model using n4sid 6 states
msleep6=n4sid(zsleepe,6,'nk',[1 1 0 0], 'DisturbanceModel','None');
[msleep6B,Ds6] = BALREAL(msleep6);
Find Awake Model using n4sid 7 states
mawake7=n4sid(zawakee,7,'nk',[1 1 0 0], 'DisturbanceModel','None');
[mawake7B,Da7] = BALREAL(mawake7);
Find Sleep Model using n4sid 7 states
msleep7=n4sid(zsleepe,7,'nk',[1 1 0 0], 'DisturbanceModel','None');
[msleep7B,Ds7] = BALREAL(msleep7);
Compare models with zsleepv and zawakev
Compare mawake4 with zawakev
compare(zawakev,mawake4);
Compare mawake5 with zawakev
compare(zawakev,mawake5);
Compare mawake6 with zawakev
compare(zawakev,mawake6);
Compare mawake7 with zawakev

105

```
compare(zawakev,mawake7);
```
Compare msleep4 with zsleepv
```
compare(zsleepv,msleep4);
```
Compare msleep5 with zsleepv
```
compare(zsleepv,msleep5);
```
Compare msleep6 with zsleepv
```
compare(zsleepv,msleep6);
```
Compare msleep7 with zsleepv
```
compare(zsleepv,msleep7);
```

Patient 1 HR

Load patient 1 data
```
load patient1-2ipt-v3.mat
```
Create data
```
z=iddata
(used_hr_data',[used_iso_data' used_EP_data' used_EVAL_data' used_LMA_data'],5);
z.inputn={'ISO';'EP';'EVAL';'LMA'};
z.outputn={'HR'};
```
Partition data
```
zawake=[z(1:732,:); z(1674:2563,:)];
zsleep=z(733:1674,:);
zawakev=zawake(1:732);
zawakee=zawake(733:1622);
zsleepe=zsleep(1:471);
zsleepv=zsleep(472:942);
```
Detrend
```
zawakee=detrend(zawakee);
zawakev=detrend(zawakev);
zsleepe=detrend(zsleepe);
zsleepv=detrend(zsleepv);
```
Find Order of Sleep
```
msleep=n4sid(zsleepe,1:15);
```
Find Order of Awake
```
mawake=n4sid(zawakee,1:15);
```

106

Impulse Response of Sleep Model
impulse(msleep,'sd',3,'fill');
Impulse Response of Awake Model
impulse(mawake,'sd',3,'fill');
Find Awake Model using n4sid 4 states
mawake4=n4sid(zawakee,4,'nk',[1 0 0 0], 'Focus','Simulation');
[mawake4B,Da4] = BALREAL(mawake4);
Find Sleep Model using n4sid 4 states
msleep4=n4sid(zsleepe,4,'nk',[1 0 0 0], 'Focus','Simulation');
[msleep4B,Ds4] = BALREAL(msleep4);
Find Awake Model using n4sid 5 states
mawake5=n4sid(zawakee,5,'nk',[1 0 0 0], 'Focus','Simulation');
[mawake5B,Da5] = BALREAL(mawake5);
Find Sleep Model using n4sid 5 states
msleep5=n4sid(zsleepe,5,'nk',[1 0 0 0], 'Focus','Simulation');
[msleep5B,Ds5] = BALREAL(msleep5);
Find Awake Model using n4sid 6 states
mawake6=n4sid(zawakee,6,'nk',[1 1 0 0], 'DisturbanceModel','None');
[mawake6B,Da6] = BALREAL(mawake6);
Find Sleep Model using n4sid 6 states
msleep6=n4sid(zsleepe,6,'nk',[1 1 0 0], 'DisturbanceModel','None');
[msleep6B,Ds6] = BALREAL(msleep6);
Find Awake Model using n4sid 7 states
mawake7=n4sid(zawakee,7,'nk',[1 1 0 0], 'DisturbanceModel','None');
[mawake7B,Da7] = BALREAL(mawake7);
Find Sleep Model using n4sid 7 states
msleep7=n4sid(zsleepe,7,'nk',[1 1 0 0], 'DisturbanceModel','None');
[msleep7B,Ds7] = BALREAL(msleep7);
Compare models with zsleepv and zawakev
Compare mawake4 with zawakev
compare(zawakev,mawake4);
Compare mawake5 with zawakev
compare(zawakev,mawake5);
Compare mawake6 with zawakev
compare(zawakev,mawake6);
Compare mawake7 with zawakev

107

```
compare(zawakev,mawake7);
```
Compare msleep4 with zsleepv
```
compare(zsleepv,msleep4);
```
Compare msleep5 with zsleepv
```
compare(zsleepv,msleep5);
```
Compare msleep6 with zsleepv
```
compare(zsleepv,msleep6);
```
Compare msleep7 with zsleepv
```
compare(zsleepv,msleep7);
```

Patient 2 BIS

Load patient 2 data
```
load patient2-2ipt-v3.mat
```
Create data
```
z=iddata
(used_bis_data',[used_iso_data' used_EP_data' used_EVAL_data' used_LMA_data'],5);
z.inputn='ISO';'EP';'EVAL';'LMA';
z.outputn='BIS';
```
Partition data
```
zawake=[z(1:903,:); z(2141:2874,:)];
zsleep=z(904:2140,:);
zawakee=zawake(1:903);
zawakev=zawake(904:1637);
zsleepe=zsleep(1:618);
zsleepv=zsleep(619:1236);
```
Detrend
```
zawakee=detrend(zawakee);
zawakev=detrend(zawakev);
zsleepe=detrend(zsleepe);
zsleepv=detrend(zsleepv);
```
Find Order of Sleep
```
msleep=n4sid(zsleepe,1:15);
```
Find Order of Awake
```
mawake=n4sid(zawakee,1:15);
```

108

Impulse Response of Sleep Model
impulse(msleep,'sd',3,'fill');
Impulse Response of Awake Model
impulse(mawake,'sd',3,'fill');
Find Awake Model using n4sid 4 states
mawake4=n4sid(zawakee,4,'nk',[1 1 0 0], 'Focus', 'Simulation');
[mawake4B,Da4] = BALREAL(mawake4);
Find Sleep Model using n4sid 4 states
msleep4=n4sid(zsleepe,4,'nk',[1 0 0 0], 'Focus', 'Stability');
[msleep4B,Ds4] = BALREAL(msleep4);
Find Awake Model using n4sid 5 states
mawake5=n4sid(zawakee,5,'nk',[1 1 0 0], 'Focus', 'Simulation');
[mawake5B,Da5] = BALREAL(mawake5);
Find Sleep Model using n4sid 5 states
msleep5=n4sid(zsleepe,5,'nk',[1 0 0 0], 'Focus', 'Stability');
[msleep5B,Ds5] = BALREAL(msleep5);
Find Awake Model using n4sid 6 states
mawake6=n4sid(zawakee,6,'nk',[1 1 0 0], 'DisturbanceModel','None');
[mawake6B,Da6] = BALREAL(mawake6);
Find Sleep Model using n4sid 6 states
msleep6=n4sid(zsleepe,6,'nk',[1 1 0 0], 'DisturbanceModel','None');
[msleep6B,Ds6] = BALREAL(msleep6);
Find Awake Model using n4sid 7 states
mawake7=n4sid(zawakee,7,'nk',[1 1 0 0], 'DisturbanceModel','None');
[mawake7B,Da7] = BALREAL(mawake7);
Find Sleep Model using n4sid 7 states
msleep7=n4sid(zsleepe,7,'nk',[1 1 0 0], 'DisturbanceModel','None');
[msleep7B,Ds7] = BALREAL(msleep7);
Compare models with zsleepv and zawakev
Compare mawake4 with zawakev
compare(zawakev,mawake4);
Compare mawake5 with zawakev
compare(zawakev,mawake5);
Compare mawake6 with zawakev
compare(zawakev,mawake6);
Compare mawake7 with zawakev

109

```
compare(zawakev,mawake7);
```
Compare msleep4 with zsleepv
```
compare(zsleepv,msleep4);
```
Compare msleep5 with zsleepv
```
compare(zsleepv,msleep5);
```
Compare msleep6 with zsleepv
```
compare(zsleepv,msleep6);
```
Compare msleep7 with zsleepv
```
compare(zsleepv,msleep7);
```


Patient 2 BP

Load patient 2 data
```
load patient2-2ipt-v3.mat
```
Create data
```
z=iddata
(used_bp_data',[used_iso_data' used_EP_data' used_EVAL_data' used_LMA_data'],5);
z.inputn='ISO';'EP';'EVAL';'LMA';
z.outputn='BP';
```
Partition data
```
zawake=[z(1:903,:); z(2141:2874,:)];
zsleep=z(904:2140,:);
zawakee=zawake(1:903);
zawakev=zawake(904:1637);
zsleepe=zsleep(1:618);
zsleepv=zsleep(619:1236);
```
Detrend
```
zawakee=detrend(zawakee);
zawakev=detrend(zawakev);
zsleepe=detrend(zsleepe);
zsleepv=detrend(zsleepv);
```
Find Order of Sleep
```
msleep=n4sid(zsleepe,1:15);
```
Find Order of Awake
```
mawake=n4sid(zawakee,1:15);
```

110

Impulse Response of Sleep Model
impulse(msleep,'sd',3,'fill');
Impulse Response of Awake Model
impulse(mawake,'sd',3,'fill');
Find Awake Model using n4sid 1 states
mawake4=n4sid(zawakee,1,'nk',[1 1 0 0], 'Focus', 'Simulation');
[mawake4B,Da4] = BALREAL(mawake4);
Find Sleep Model using n4sid 1 states
msleep4=n4sid(zsleepe,1,'nk',[1 0 0 0], 'Focus', 'Stability');
[msleep4B,Ds4] = BALREAL(msleep4);
Find Awake Model using n4sid 2 states
mawake5=n4sid(zawakee,2,'nk',[1 1 0 0], 'Focus', 'Simulation');
[mawake5B,Da5] = BALREAL(mawake5);
Find Sleep Model using n4sid 2 states
msleep5=n4sid(zsleepe,2,'nk',[1 0 0 0], 'Focus', 'Stability');
[msleep5B,Ds5] = BALREAL(msleep5);
Find Awake Model using n4sid 6 states
mawake6=n4sid(zawakee,6,'nk',[1 1 0 0], 'DisturbanceModel','None');
[mawake6B,Da6] = BALREAL(mawake6);
Find Sleep Model using n4sid 6 states
msleep6=n4sid(zsleepe,6,'nk',[1 1 0 0], 'DisturbanceModel','None');
[msleep6B,Ds6] = BALREAL(msleep6);
Find Awake Model using n4sid 7 states
mawake7=n4sid(zawakee,7,'nk',[1 1 0 0], 'DisturbanceModel','None');
[mawake7B,Da7] = BALREAL(mawake7);
Find Sleep Model using n4sid 7 states
msleep7=n4sid(zsleepe,7,'nk',[1 1 0 0], 'DisturbanceModel','None');
[msleep7B,Ds7] = BALREAL(msleep7);
Compare models with zsleepv and zawakev
Compare mawake4 with zawakev
compare(zawakev,mawake4);
Compare mawake5 with zawakev
compare(zawakev,mawake5);
Compare mawake6 with zawakev
compare(zawakev,mawake6);
Compare mawake7 with zawakev

111

```
compare(zawakev,mawake7);
```
Compare msleep4 with zsleepv
```
compare(zsleepv,msleep4);
```
Compare msleep5 with zsleepv
```
compare(zsleepv,msleep5);
```
Compare msleep6 with zsleepv
```
compare(zsleepv,msleep6);
```
Compare msleep7 with zsleepv
```
compare(zsleepv,msleep7);
```

A.1.1   Patient 3 BIS

Load patient 3 data
```
load patient3-2ipt-v3.mat
```
Create data
```
z=iddata
(used_bis_data',[used_iso_data' used_EP_data' used_EVAL_data' used_LMA_data'],5);
z.inputn='ISO';'EP';'EVAL';'LMA';
z.outputn='BIS';
```
Partition data
```
zawake=[z(1:913,:); z(2279:2678,:)];
zsleep=z(914:2278,:);
zawakee=zawake(1:913);
zawakev=zawake(914:2751);
zsleepe=zsleep(1:682);
zsleepv=zsleep(683:1364);
```
Detrend
```
zawakee=detrend(zawakee);
zawakev=detrend(zawakev);
zsleepe=detrend(zsleepe);
zsleepv=detrend(zsleepv);
```
Find Order of Sleep
```
msleep=n4sid(zsleepe,1:15);
```
Find Order of Awake
```
mawake=n4sid(zawakee,1:15);
```

112

Impulse Response of Sleep Model

impulse(msleep,'sd',3,'fill');

Impulse Response of Awake Model

impulse(mawake,'sd',3,'fill');

Find Awake Model using n4sid 4 states

mawake4=n4sid(zawakee,4,'nk',[1 0 0 0], 'Focus','Simulation');

[mawake4B,Da4] = BALREAL(mawake4);

Find Sleep Model using n4sid 4 states

msleep4=n4sid(zsleepe,4,'nk',[1 0 0 0], 'Focus','Simulation');

[msleep4B,Ds4] = BALREAL(msleep4);

Find Awake Model using n4sid 5 states

mawake5=n4sid(zawakee,5,'nk',[1 0 0 0], 'Focus','Simulation');

[mawake5B,Da5] = BALREAL(mawake5);

Find Sleep Model using n4sid 5 states

msleep5=n4sid(zsleepe,5,'nk',[1 0 0 0], 'Focus','Simulation');

[msleep5B,Ds5] = BALREAL(msleep5);

Find Awake Model using n4sid 6 states

mawake6=n4sid(zawakee,6,'nk',[1 1 0 0], 'DisturbanceModel','None');

[mawake6B,Da6] = BALREAL(mawake6);

Find Sleep Model using n4sid 6 states

msleep6=n4sid(zsleepe,6,'nk',[1 1 0 0], 'DisturbanceModel','None');

[msleep6B,Ds6] = BALREAL(msleep6);

Find Awake Model using n4sid 7 states

mawake7=n4sid(zawakee,7,'nk',[1 1 0 0], 'DisturbanceModel','None');

[mawake7B,Da7] = BALREAL(mawake7);

Find Sleep Model using n4sid 7 states

msleep7=n4sid(zsleepe,7,'nk',[1 1 0 0], 'DisturbanceModel','None');

[msleep7B,Ds7] = BALREAL(msleep7);

Compare models with zsleepv and zawakev

Compare mawake4 with zawakev

compare(zawakev,mawake4);

Compare mawake5 with zawakev

compare(zawakev,mawake5);

Compare mawake6 with zawakev

compare(zawakev,mawake6);

Compare mawake7 with zawakev

113

```
compare(zawakev,mawake7);
```
Compare msleep4 with zsleepv
```
compare(zsleepv,msleep4);
```
Compare msleep5 with zsleepv
```
compare(zsleepv,msleep5);
```
Compare msleep6 with zsleepv
```
compare(zsleepv,msleep6);
```
Compare msleep7 with zsleepv
```
compare(zsleepv,msleep7);
```

Patient 3 BP

Load patient 3 data
```
load patient3-2ipt-v3.mat
```
Create data
```
z=iddata
(used bp data',[used iso data' used EP data' used EVAL data' used LMA data'],5);
z.inputn='ISO';'EP';'EVAL';'LMA';
z.outputn='BP';
```
Partition data
```
zawake=[z(1:78,:); z(82:430,:); z(433:913,:); z(1947:1963,:);
z(2279:2725,:); z(2729:2751,:)];
zsleep=[z(79:82,:);z(431:432,:); z(914:1946,:); z(1964:2278,:);
z(2726:2728,:)];
zawakee=zawake(1:913);
zawakev=zawake(914:2751);
zsleepe=zsleep(1:674);
zsleepv=zsleep(675:1349);
```
Detrend
```
zawakee=detrend(zawakee);
zawakev=detrend(zawakev);
zsleepe=detrend(zsleepe);
zsleepv=detrend(zsleepv);
```
Find Order of Sleep
```
msleep=n4sid(zsleepe,1:15);
```

114

Find Order of Awake

mawake=n4sid(zawakee,1:15);

Impulse Response of Sleep Model

impulse(msleep,'sd',3,'fill');

Impulse Response of Awake Model

impulse(mawake,'sd',3,'fill');

Find Awake Model using n4sid 4 states

mawake4=n4sid(zawakee,1,'nk',[1 0 0 0], 'Focus','Simulation');

[mawake4B,Da4] = BALREAL(mawake4);

Find Sleep Model using n4sid 4 states

msleep4=n4sid(zsleepv,1,'nk',[1 0 0 0], 'Focus','Simulation');

[msleep4B,Ds4] = BALREAL(msleep4);

Find Awake Model using n4sid 5 states

mawake5=n4sid(zawakee,2,'nk',[1 0 0 0], 'Focus','Simulation');

[mawake5B,Da5] = BALREAL(mawake5);

Find Sleep Model using n4sid 5 states

msleep5=n4sid(zsleepv,2,'nk',[1 0 0 0], 'Focus','Simulation');

[msleep5B,Ds5] = BALREAL(msleep5);

Find Awake Model using n4sid 6 states

mawake6=n4sid(zawakee,6,'nk',[1 1 0 0], 'DisturbanceModel','None');

[mawake6B,Da6] = BALREAL(mawake6);

Find Sleep Model using n4sid 6 states

msleep6=n4sid(zsleepe,6,'nk',[1 1 0 0], 'DisturbanceModel','None');

[msleep6B,Ds6] = BALREAL(msleep6);

Find Awake Model using n4sid 7 states

mawake7=n4sid(zawakee,7,'nk',[1 1 0 0], 'DisturbanceModel','None');

[mawake7B,Da7] = BALREAL(mawake7);

Find Sleep Model using n4sid 7 states

msleep7=n4sid(zsleepe,7,'nk',[1 1 0 0], 'DisturbanceModel','None');

[msleep7B,Ds7] = BALREAL(msleep7);

Compare models with zsleepv and zawakev

Compare mawake4 with zawakev

compare(zawakev,mawake4);

Compare mawake5 with zawakev

compare(zawakev,mawake5);

Compare mawake6 with zawakev

115

```
    compare(zawakev,mawake6);
```
Compare mawake7 with zawakev
```
    compare(zawakev,mawake7);
```
Compare msleep4 with zsleepv
```
    compare(zsleepv,msleep4);
```
Compare msleep5 with zsleepv
```
    compare(zsleepv,msleep5);
```
Compare msleep6 with zsleepv
```
    compare(zsleepv,msleep6);
```
Compare msleep7 with zsleepv
```
    compare(zsleepv,msleep7);
```

Patient 5 BIS

Load patient 5 data
```
    load patient5-2ipt-v3.mat
```
Create data
```
    z=iddata
    (used_bis_data',[used_iso_data' used_EP_data' used_EVAL_data' used_LMA_data'],5);
    z.inputn='ISO';'EP';'EVAL';'LMA';
    z.outputn='BIS';
```
Partition data
```
    zawake=[z(1:660,:); z(1380:1458,:); z(1644:2195,:)];
    zsleep=[z(661:1379,:); z(1459:1643,:)];
    zawakee=zawake(1:660);
    zawakev=zawake(661:1291);
    zsleepe=zsleep(1:452);
    zsleepv=zsleep(453:904);
```
Detrend
```
    zawakee=detrend(zawakee);
    zawakev=detrend(zawakev);
    zsleepe=detrend(zsleepe);
    zsleepv=detrend(zsleepv);
```
Find Order of Sleep
```
    msleep=n4sid(zsleepe,1:15);
```

116

Find Order of Awake

mawake=n4sid(zawakee,1:15);

Impulse Response of Sleep Model

impulse(msleep,'sd',3,'fill');

Impulse Response of Awake Model

impulse(mawake,'sd',3,'fill');

Find Awake Model using n4sid 4 states

mawake4=n4sid(zawakee,4,'nk',[1 0 0 0], 'Focus','Simulation');

[mawake4B,Da4] = BALREAL(mawake4);

Find Sleep Model using n4sid 4 states

msleep4=n4sid(zsleepe,4,'nk',[1 0 0 0], 'Focus','Simulation');

[msleep4B,Ds4] = BALREAL(msleep4);

Find Awake Model using n4sid 5 states

mawake5=n4sid(zawakee,5,'nk',[1 0 0 0], 'Focus','Simulation');

[mawake5B,Da5] = BALREAL(mawake5);

Find Sleep Model using n4sid 5 states

msleep5=n4sid(zsleepe,5,'nk',[1 0 0 0], 'Focus','Simulation');

[msleep5B,Ds5] = BALREAL(msleep5);

Find Awake Model using n4sid 6 states

mawake6=n4sid(zawakee,6,'nk',[1 1 0 0], 'DisturbanceModel','None');

[mawake6B,Da6] = BALREAL(mawake6);

Find Sleep Model using n4sid 6 states

msleep6=n4sid(zsleepe,6,'nk',[1 1 0 0], 'DisturbanceModel','None');

[msleep6B,Ds6] = BALREAL(msleep6);

Find Awake Model using n4sid 7 states

mawake7=n4sid(zawakee,7,'nk',[1 1 0 0], 'DisturbanceModel','None');

[mawake7B,Da7] = BALREAL(mawake7);

Find Sleep Model using n4sid 7 states

msleep7=n4sid(zsleepe,7,'nk',[1 1 0 0], 'DisturbanceModel','None');

[msleep7B,Ds7] = BALREAL(msleep7);

Compare models with zsleepv and zawakev

Compare mawake4 with zawakev

compare(zawakev,mawake4);

Compare mawake5 with zawakev

compare(zawakev,mawake5);

Compare mawake6 with zawakev

117

```
    compare(zawakev,mawake6);
```
Compare mawake7 with zawakev
```
    compare(zawakev,mawake7);
```
Compare msleep4 with zsleepv
```
    compare(zsleepv,msleep4);
```
Compare msleep5 with zsleepv
```
    compare(zsleepv,msleep5);
```
Compare msleep6 with zsleepv
```
    compare(zsleepv,msleep6);
```
Compare msleep7 with zsleepv
```
    compare(zsleepv,msleep7);
```


Patient 5 BP

Load patient 5 data
```
    load patient5-2ipt-v3.mat
```
Create data
```
    z=iddata
    (used_bp_data',[used_iso_data' used_EP_data' used_EVAL_data' used_LMA_data'],5);
    z.inputn='ISO';'EP';'EVAL';'LMA';
    z.outputn='BP';
```
Partition data
```
    zawake=[z(1:660,:); z(1380:1458,:); z(1644:2195,:)];
    zsleep=[z(661:1379,:); z(1459:1643,:)];
    zawakee=zawake(1:660);
    zawakev=zawake(661:1291);
    zsleepe=zsleep(1:452);
    zsleepv=zsleep(453:904);
```
Detrend
```
    zawakee=detrend(zawakee);
    zawakev=detrend(zawakev);
    zsleepe=detrend(zsleepe);
    zsleepv=detrend(zsleepv);
```
Find Order of Sleep
```
    msleep=n4sid(zsleepe,1:15);
```

118

Find Order of Awake

mawake=n4sid(zawakee,1:15);

Impulse Response of Sleep Model

impulse(msleep,'sd',3,'fill');

Impulse Response of Awake Model

impulse(mawake,'sd',3,'fill');

Find Awake Model using n4sid 4 states

mawake4=n4sid(zawakee,1,'nk',[1 0 0 0], 'Focus','Simulation');

[mawake4B,Da4] = BALREAL(mawake4);

Find Sleep Model using n4sid 4 states

msleep4=n4sid(zsleepe,1,'nk',[1 0 0 0], 'Focus','Stability');

[msleep4B,Ds4] = BALREAL(msleep4);

Find Awake Model using n4sid 5 states

mawake5=n4sid(zawakee,2,'nk',[1 0 0 0], 'Focus','Simulation');

[mawake5B,Da5] = BALREAL(mawake5);

Find Sleep Model using n4sid 5 states

msleep5=n4sid(zsleepe,2,'nk',[1 0 0 0], 'Focus','Stability');

[msleep5B,Ds5] = BALREAL(msleep5);

Find Awake Model using n4sid 6 states

mawake6=n4sid(zawakee,6,'nk',[1 1 0 0], 'DisturbanceModel','None');

[mawake6B,Da6] = BALREAL(mawake6);

Find Sleep Model using n4sid 6 states

msleep6=n4sid(zsleepe,6,'nk',[1 1 0 0], 'DisturbanceModel','None');

[msleep6B,Ds6] = BALREAL(msleep6);

Find Awake Model using n4sid 7 states

mawake7=n4sid(zawakee,7,'nk',[1 1 0 0], 'DisturbanceModel','None');

[mawake7B,Da7] = BALREAL(mawake7);

Find Sleep Model using n4sid 7 states

msleep7=n4sid(zsleepe,7,'nk',[1 1 0 0], 'DisturbanceModel','None');

[msleep7B,Ds7] = BALREAL(msleep7);

Compare models with zsleepv and zawakev

Compare mawake4 with zawakev

compare(zawakev,mawake4);

Compare mawake5 with zawakev

compare(zawakev,mawake5);

Compare mawake6 with zawakev

119

```
    compare(zawakev,mawake6);
```
Compare mawake7 with zawakev
```
    compare(zawakev,mawake7);
```
Compare msleep4 with zsleepv
```
    compare(zsleepv,msleep4);
```
Compare msleep5 with zsleepv
```
    compare(zsleepv,msleep5);
```
Compare msleep6 with zsleepv
```
    compare(zsleepv,msleep6);
```
Compare msleep7 with zsleepv
```
    compare(zsleepv,msleep7);
```


Patient 6 BIS

Load patient 6 data
```
    load patient6-2ipt-v3.mat
```
Create data
```
    z=iddata
    (used_bis_data',[used_iso_data' used_EP_data' used_EVAL_data' used_LMA_data'],5);
    z.inputn='ISO';'EP';'EVAL';'LMA';
    z.outputn='BIS';
```
Partition data
```
    zawake=[z(1:68,:); z(88:130,:); z(139:236,:); z(240:257,:); z(265:286,:); z(293:303,:);
z(306:354,:); z(371:380,:); z(394:764,:); z(773:780,:); z(874:939,:); z(944:1000,:);
z(1007:1022,:); z(1028:1085,:); z(1153:1159,:); z(1191:1195,:); z(1205:1211,:);
z(1251:1251,:); z(1579:1581,:); z(1598:1601,:); z(1616:1624,:); z(1634:1635,:);
z(1686:1755,:); z(2062:2079,:); z(2092:2768,:); z(2778:2826,:)]; zsleep=[z(69:87,:);
z(131:138,:); z(237:239,:);z(258:264,:); z(287:292,:); z(304:305,:); z(355:370,:);
z(381:393,:); z(765:772,:); z(781:873,:); z(940:943,:); z(1001:1006,:); z(1023:1027,:);
z(1086:1152,:); z(1160:1190,:); z(1196:1204,:); z(1212:1250,:); z(1252:1578,:);
z(1582:1597,:); z(1602:1615,:); z(1625:1633,:); z(1636:1685,:); z(1756:2061,:);
z(2080:2091,:); z(2769:2777,:)];
    zawakee=zawake(1:873);
    zawakev=zawake(874:1747);
    zsleepe=zsleep(1:539);
```

120

```
zsleepv=zsleep(540:1079);
```
Detrend
```
zawakee=detrend(zawakee);
zawakev=detrend(zawakev);
zsleepe=detrend(zsleepe);
zsleepv=detrend(zsleepv);
```
Find Order of Sleep
```
msleep=n4sid(zsleepe,1:15);
```
Find Order of Awake
```
mawake=n4sid(zawakee,1:15);
```
Impulse Response of Sleep Model
```
impulse(msleep,'sd',3,'fill');
```
Impulse Response of Awake Model
```
impulse(mawake,'sd',3,'fill');
```
Find Awake Model using n4sid 4 states
```
mawake4=n4sid(zawakee,4,'nk',[2 0 0 0], 'Focus', 'Stability' )
[mawake4B,Da4] = BALREAL(mawake4);
```
Find Sleep Model using n4sid 4 states
```
msleep4=n4sid(zsleepe,4,'nk',[2 0 0 0], 'Focus', 'Stability' )
[msleep4B,Ds4] = BALREAL(msleep4);
```
Find Awake Model using n4sid 5 states
```
mawake5=n4sid(zawakee,5,'nk',[2 0 0 0], 'Focus', 'Stability' )
[mawake5B,Da5] = BALREAL(mawake5);
```
Find Sleep Model using n4sid 5 states
```
msleep5=n4sid(zsleepe,5,'nk',[2 0 0 0], 'Focus', 'Stability' )
[msleep5B,Ds5] = BALREAL(msleep5);
```
Find Awake Model using n4sid 6 states
```
mawake6=n4sid(zawakee,6,'nk',[1 1 0 0], 'DisturbanceModel','None');
[mawake6B,Da6] = BALREAL(mawake6);
```
Find Sleep Model using n4sid 6 states
```
msleep6=n4sid(zsleepe,6,'nk',[1 1 0 0], 'DisturbanceModel','None');
[msleep6B,Ds6] = BALREAL(msleep6);
```
Find Awake Model using n4sid 7 states
```
mawake7=n4sid(zawakee,7,'nk',[1 1 0 0], 'DisturbanceModel','None');
[mawake7B,Da7] = BALREAL(mawake7);
```
Find Sleep Model using n4sid 7 states

121

```
msleep7=n4sid(zsleepe,7,'nk',[1 1 0 0], 'DisturbanceModel','None');
[msleep7B,Ds7] = BALREAL(msleep7);
```
Compare models with zsleepv and zawakev
Compare mawake4 with zawakev
```
compare(zawakev,mawake4);
```
Compare mawake5 with zawakev
```
compare(zawakev,mawake5);
```
Compare mawake6 with zawakev
```
compare(zawakev,mawake6);
```
Compare mawake7 with zawakev
```
compare(zawakev,mawake7);
```
Compare msleep4 with zsleepv
```
compare(zsleepv,msleep4);
```
Compare msleep5 with zsleepv
```
compare(zsleepv,msleep5);
```
Compare msleep6 with zsleepv
```
compare(zsleepv,msleep6);
```
Compare msleep7 with zsleepv
```
compare(zsleepv,msleep7);
```

Patient 6 BP

Load patient 6 data
```
load patient6-2ipt-v3.mat
```
Create data
```
z=iddata
(used_bp_data',[used_iso_data' used_EP_data' used_EVAL_data' used_LMA_data'],5);
z.inputn='ISO';'EP';'EVAL';'LMA';
z.outputn='BP';
```
Partition data
```
zawake=[z(1:68,:); z(88:130,:); z(139:236,:); z(240:257,:); z(265:286,:); z(293:303,:);
z(306:354,:); z(371:380,:); z(394:764,:); z(773:780,:); z(874:939,:); z(944:1000,:);
z(1007:1022,:); z(1028:1085,:); z(1153:1159,:); z(1191:1195,:); z(1205:1211,:);
z(1251:1251,:); z(1579:1581,:); z(1598:1601,:); z(1616:1624,:); z(1634:1635,:);
z(1686:1755,:); z(2062:2079,:); z(2092:2768,:); z(2778:2826,:)]; zsleep=[z(69:87,:);
```

```
z(131:138,:); z(237:239,:);z(258:264,:); z(287:292,:); z(304:305,:); z(355:370,:);
z(381:393,:); z(765:772,:); z(781:873,:); z(940:943,:); z(1001:1006,:); z(1023:1027,:);
z(1086:1152,:); z(1160:1190,:); z(1196:1204,:); z(1212:1250,:); z(1252:1578,:);
z(1582:1597,:); z(1602:1615,:); z(1625:1633,:); z(1636:1685,:); z(1756:2061,:);
z(2080:2091,:); z(2769:2777,:)];
  zawakee=zawake(1:873);
  zawakev=zawake(874:1747);
  zsleepe=zsleep(1:539);
  zsleepv=zsleep(540:1079);
  Detrend
  zawakee=detrend(zawakee);
  zawakev=detrend(zawakev);
  zsleepe=detrend(zsleepe);
  zsleepv=detrend(zsleepv);
  Find Order of Sleep
  msleep=n4sid(zsleepe,1:15);
  Find Order of Awake
  mawake=n4sid(zawakee,1:15);
  Impulse Response of Sleep Model
  impulse(msleep,'sd',3,'fill');
  Impulse Response of Awake Model
  impulse(mawake,'sd',3,'fill');
  Find Awake Model using n4sid 4 states
  mawake4=n4sid(zawakee,1,'nk',[2 0 0 0], 'Focus', 'Stability' )
  [mawake4B,Da4] = BALREAL(mawake4);
  Find Sleep Model using n4sid 4 states
  msleep4=n4sid(zsleepe,1,'nk',[2 0 0 0], 'Focus', 'Stability' )
  [msleep4B,Ds4] = BALREAL(msleep4);
  Find Awake Model using n4sid 5 states
  mawake5=n4sid(zawakee,2,'nk',[2 0 0 0], 'Focus', 'Stability' )
  [mawake5B,Da5] = BALREAL(mawake5);
  Find Sleep Model using n4sid 5 states
  msleep5=n4sid(zsleepe,2,'nk',[2 0 0 0], 'Focus', 'Stability' )
  [msleep5B,Ds5] = BALREAL(msleep5);
  Find Awake Model using n4sid 6 states
  mawake6=n4sid(zawakee,6,'nk',[1 1 0 0], 'DisturbanceModel','None');
```

123

```
[mawake6B,Da6] = BALREAL(mawake6);
```
Find Sleep Model using n4sid 6 states
```
msleep6=n4sid(zsleepe,6,'nk',[1 1 0 0], 'DisturbanceModel','None');
[msleep6B,Ds6] = BALREAL(msleep6);
```
Find Awake Model using n4sid 7 states
```
mawake7=n4sid(zawakee,7,'nk',[1 1 0 0], 'DisturbanceModel','None');
[mawake7B,Da7] = BALREAL(mawake7);
```
Find Sleep Model using n4sid 7 states
```
msleep7=n4sid(zsleepe,7,'nk',[1 1 0 0], 'DisturbanceModel','None');
[msleep7B,Ds7] = BALREAL(msleep7);
```
Compare models with zsleepv and zawakev

Compare mawake4 with zawakev
```
compare(zawakev,mawake4);
```
Compare mawake5 with zawakev
```
compare(zawakev,mawake5);
```
Compare mawake6 with zawakev
```
compare(zawakev,mawake6);
```
Compare mawake7 with zawakev
```
compare(zawakev,mawake7);
```
Compare msleep4 with zsleepv
```
compare(zsleepv,msleep4);
```
Compare msleep5 with zsleepv
```
compare(zsleepv,msleep5);
```
Compare msleep6 with zsleepv
```
compare(zsleepv,msleep6);
```
Compare msleep7 with zsleepv
```
compare(zsleepv,msleep7);
```

Patient 7 BIS

Load patient 7 data
```
load patient7-2ipt-v3.mat
```
Create data
```
z=iddata
(used_bis_data',[used_iso_data' used_EP_data' used_EVAL_data' used_LMA_data'],5);
```

124

```
z.inputn='ISO';'EP';'EVAL';'LMA';
z.outputn='BIS';
```
Partition data
```
zawake=[z(319:346,:); z(367:414,:);z(434:460,:);z(495:507,:);
z(513:1140,:);z(1152:1166,:);z(2338:2367,:) ;z(2637:2990,:)];
zsleep=[z(1167:2337,:);z(2368:2636,:)];
zawakev=zawake(1:571);
zawakee=zawake(572:1143);
zsleepe=zsleep(1:720);
zsleepv=zsleep(721:1440);
```
Detrend
```
zawakee=detrend(zawakee);
zawakev=detrend(zawakev);
zsleepe=detrend(zsleepe);
zsleepv=detrend(zsleepv);
```
Find Order of Sleep
```
msleep=n4sid(zsleepe,1:15);
```
Find Order of Awake
```
mawake=n4sid(zawakee,1:15);
```
Impulse Response of Sleep Model
```
impulse(msleep,'sd',3,'fill');
```
Impulse Response of Awake Model
```
impulse(mawake,'sd',3,'fill');
```
Find Awake Model using n4sid 4 states
```
mawake4=n4sid(zawakee,4,'nk',[1 1 0 0],'Focus','Simulation')
[mawake4B,Da4] = BALREAL(mawake4);
```
Find Sleep Model using n4sid 4 states
```
msleep4=n4sid(zsleepe,4,'nk',[1 1 0 0],'Focus','Simulation')
[msleep4B,Ds4] = BALREAL(msleep4);
```
Find Awake Model using n4sid 5 states
```
mawake5=n4sid(zawakee,5,'nk',[1 1 0 0],'Focus','Simulation')
[mawake5B,Da5] = BALREAL(mawake5);
```
Find Sleep Model using n4sid 5 states
```
msleep5=n4sid(zsleepe,5,'nk',[1 1 0 0],'Focus','Simulation')
[msleep5B,Ds5] = BALREAL(msleep5);
```
Find Awake Model using n4sid 6 states
```
mawake6=n4sid(zawakee,6,'nk',[0 0 0 0],'Focus','Stability')
```

125

```
[mawake6B,Da6] = BALREAL(mawake6);
```
Find Sleep Model using n4sid 6 states
```
msleep6=n4sid(zsleepe,6,'nk',[0 0 0 0],'Focus','Stability')
[msleep6B,Ds6] = BALREAL(msleep6);
```
Find Awake Model using n4sid 7 states
```
mawake7=n4sid(zawakee,7,'nk',[0 0 0 0],'Focus','Stability')
[mawake7B,Da7] = BALREAL(mawake7);
```
Find Sleep Model using n4sid 7 states
```
msleep7=n4sid(zsleepe,7,'nk',[0 0 0 0],'Focus','Stability')
[msleep7B,Ds7] = BALREAL(msleep7);
```
Compare models with zsleepv and zawakev
Compare mawake4 with zawakev
```
compare(zawakev,mawake4);
```
Compare mawake5 with zawakev
```
compare(zawakev,mawake5);
```
Compare mawake6 with zawakev
```
compare(zawakev,mawake6);
```
Compare mawake7 with zawakev
```
compare(zawakev,mawake7);
```
Compare msleep4 with zsleepv
```
compare(zsleepv,msleep4);
```
Compare msleep5 with zsleepv
```
compare(zsleepv,msleep5);
```
Compare msleep6 with zsleepv
```
compare(zsleepv,msleep6);
```
Compare msleep7 with zsleepv
```
compare(zsleepv,msleep7);
```

Patient 7 BP

Load patient 7 data
```
load patient7-2ipt-v3.mat
```
Create data
```
z=iddata
(used_bp_data',[used_iso_data' used_EP_data' used_EVAL_data' used_LMA_data'],5);
```

126

```
z.inputn='ISO';'EP';'EVAL';'LMA';
z.outputn='BP';
Partition data
zawake=[z(319:346,:);z(367:414,:);z(434:460,:);z(495:507,:);
z(513:1140,:);z(1152:1166,:);z(2338:2367,:) ;z(2637:2990,:)];
zsleep=[z(1167:2337,:);z(2368:2636,:)];
zawakev=zawake(1:571);
zawakee=zawake(572:1143);
zsleepe=zsleep(1:720);
zsleepv=zsleep(721:1440);
Detrend
zawakee=detrend(zawakee);
zawakev=detrend(zawakev);
zsleepe=detrend(zsleepe);
zsleepv=detrend(zsleepv);
Find Order of Sleep
msleep=n4sid(zsleepe,1:15);
Find Order of Awake
mawake=n4sid(zawakee,1:15);
Impulse Response of Sleep Model
impulse(msleep,'sd',3,'fill');
Impulse Response of Awake Model
impulse(mawake,'sd',3,'fill');
Find Awake Model using n4sid 4 states
mawake4=n4sid(zawakee,1,'nk',[1 1 0 0],'Focus','Simulation')
[mawake4B,Da4] = BALREAL(mawake4);
Find Sleep Model using n4sid 4 states
msleep4=n4sid(zsleepv,1,'nk',[1 1 0 0],'Focus','Simulation')
[msleep4B,Ds4] = BALREAL(msleep4);
Find Awake Model using n4sid 5 states
mawake5=n4sid(zawakee,2,'nk',[1 1 0 0],'Focus','Simulation')
[mawake5B,Da5] = BALREAL(mawake5);
Find Sleep Model using n4sid 5 states
msleep5=n4sid(zsleepv,2,'nk',[1 1 0 0],'Focus','Simulation')
[msleep5B,Ds5] = BALREAL(msleep5);
Find Awake Model using n4sid 6 states
```

```
mawake6=n4sid(zawakee,6,'nk',[0 0 0 0],'Focus','Stability')
[mawake6B,Da6] = BALREAL(mawake6);
```
Find Sleep Model using n4sid 6 states
```
msleep6=n4sid(zsleepe,6,'nk',[0 0 0 0],'Focus','Stability')
[msleep6B,Ds6] = BALREAL(msleep6);
```
Find Awake Model using n4sid 7 states
```
mawake7=n4sid(zawakee,7,'nk',[0 0 0 0],'Focus','Stability')
[mawake7B,Da7] = BALREAL(mawake7);
```
Find Sleep Model using n4sid 7 states
```
msleep7=n4sid(zsleepe,7,'nk',[0 0 0 0],'Focus','Stability')
[msleep7B,Ds7] = BALREAL(msleep7);
```
Compare models with zsleepv and zawakev

Compare mawake4 with zawakev
```
compare(zawakev,mawake4);
```
Compare mawake5 with zawakev
```
compare(zawakev,mawake5);
```
Compare mawake6 with zawakev
```
compare(zawakev,mawake6);
```
Compare mawake7 with zawakev
```
compare(zawakev,mawake7);
```
Compare msleep4 with zsleepv
```
compare(zsleepv,msleep4);
```
Compare msleep5 with zsleepv
```
compare(zsleepv,msleep5);
```
Compare msleep6 with zsleepv
```
compare(zsleepv,msleep6);
```
Compare msleep7 with zsleepv
```
compare(zsleepv,msleep7);
```

A.2   idgrey

BIS

Patient 7 on Patient 1
```
m1=idgrey('patient',[0;0;0;0],'c');
mp1sleep=pem(m1,zsleepe);
```

```
A1=mp1sleep.a;
B1=mp1sleep.b;
C1=mp1sleep.c;
D1=mp1sleep.d;
```
Patient 7 on Patient 2
```
m2=idgrey('patient',[0;0;0;0],'c');
mp2sleep=pem(m2,zsleepe)
A2=mp2sleep.a;
B2=mp2sleep.b;
C2=mp2sleep.c;
D2=mp2sleep.d;
```
Patient 7 on Patient 5
```
m5=idgrey('patient',[0;0;0;0],'c');
mp5sleep=pem(m5,zsleepe);
A5=mp5sleep.a;
B5=mp5sleep.b;
C5=mp5sleep.c;
D5=mp5sleep.d;
```
Patient 7 on Patient 6
```
m6=idgrey('patient',[0;0;0;0],'c');
mp6sleep=pem(m6,zsleepe)
A6=mp6sleep.a;
B6=mp6sleep.b;
C6=mp6sleep.c;
D6=mp6sleep.d;
```
Patient 7 on Patient 3
```
m3=idgrey('patient',[0;0;0;0],'c');
mp3sleep=pem(m3,zsleepe)
A3=mp3sleep.a;
B3=mp3sleep.b;
C3=mp3sleep.c;
D3=mp3sleep.d;
```
Patient 5 on Patient 1
```
m1=idgrey('patient5',[0;0;0;0],'c');
mp1sleep=pem(m1,zsleepe);
A1=mp1sleep.a;
```

129

```
B1=mp1sleep.b;
C1=mp1sleep.c;
D1=mp1sleep.d;
```
Patient 5 on Patient 2
```
m2=idgrey('patient5',[0;0;0;0],'c');
mp2sleep=pem(m2,zsleepe)
A2=mp2sleep.a;
B2=mp2sleep.b;
C2=mp2sleep.c;
D2=mp2sleep.d;
```
Patient 5 on Patient 6
```
m6=idgrey('patient5',[0;0;0;0],'c');
mp6sleep=pem(m6,zsleepe)
A6=mp6sleep.a;
B6=mp6sleep.b;
C6=mp6sleep.c;
D6=mp6sleep.d;
```
Patient 5 on Patient 7
```
m7=idgrey('patient5',[0;0;0;0],'c');
mp7sleep=pem(m7,zsleepe)
A7=mp7sleep.a;
B7=mp7sleep.b;
C7=mp7sleep.c;
D7=mp7sleep.d;
```
Patient 5 on Patient 3
```
m3=idgrey('patient5',[0;0;0;0],'c');
mp3sleep=pem(m3,zsleepe)
A3=mp3sleep.a;
B3=mp3sleep.b;
C3=mp3sleep.c;
D3=mp3sleep.d;
```
Patient 1 on Patient 7
```
m7=idgrey('patient1',[0;0;0;0],'c');
mp7sleep=pem(m7,zsleepe)
A7=mp7sleep.a;
B7=mp7sleep.b;
```

130

```
C7=mp7sleep.c;
D7=mp7sleep.d;
```
Patient 1 on Patient 6
```
m6=idgrey('patient1',[0;0;0;0],'c');
mp6sleep=pem(m6,zsleepe)
A6=mp6sleep.a;
B6=mp6sleep.b;
C6=mp6sleep.c;
D6=mp6sleep.d;
```
Patient 1 on Patient 5
```
m5=idgrey('patient1',[0;0;0;0],'c');
mp5sleep=pem(m5,zsleepe)
A5=mp5sleep.a;
B5=mp5sleep.b;
C5=mp5sleep.c;
D5=mp5sleep.d;
```
Patient 1 on Patient 2
```
m2=idgrey('patient1',[0;0;0;0],'c');
mp2sleep=pem(m2,zsleepe)
A2=mp2sleep.a;
B2=mp2sleep.b;
C2=mp2sleep.c;
D2=mp2sleep.d;
```
Patient 1 on Patient 3
```
m3=idgrey('patient1',[0;0;0;0],'c');
mp3sleep=pem(m3,zsleepe)
A3=mp3sleep.a;
B3=mp3sleep.b;
C3=mp3sleep.c;
D3=mp3sleep.d;
```
Patient 6 on Patient 1
```
m1=idgrey('patient6',[0;0;0;0],'c');
mp1sleep=pem(m1,zsleepe);
A1=mp1sleep.a;
B1=mp1sleep.b;
C1=mp1sleep.c;
```

131

```
D1=mp1sleep.d;
```
Patient 6 on Patient 5
```
m5=idgrey('patient6',[0;0;0;0],'c');
mp5sleep=pem(m5,zsleepe)
A5=mp5sleep.a;
B5=mp5sleep.b;
C5=mp5sleep.c;
D5=mp5sleep.d;
```
Patient 6 on Patient 7
```
m7=idgrey('patient6',[0;0;0;0],'c');
mp7sleep=pem(m7,zsleepe)
A7=mp7sleep.a;
B7=mp7sleep.b;
C7=mp7sleep.c;
D7=mp7sleep.d;
```
Patient 6 on Patient 2
```
m2=idgrey('patient6',[0;0;0;0],'c');
mp2sleep=pem(m2,zsleepe)
A2=mp2sleep.a;
B2=mp2sleep.b;
C2=mp2sleep.c;
D2=mp2sleep.d;
```
Patient 6 on Patient 3
```
m3=idgrey('patient6',[0;0;0;0],'c');
mp3sleep=pem(m3,zsleepe)
A3=mp3sleep.a;
B3=mp3sleep.b;
C3=mp3sleep.c;
D3=mp3sleep.d;
```
Patient 3 on Patient 1
```
m1=idgrey('patient3',[0;0;0;0],'c');
mp1sleep=pem(m1,zsleepe);
A1=mp1sleep.a;
B1=mp1sleep.b;
C1=mp1sleep.c;
D1=mp1sleep.d;
```
Patient 3 on Patient 5

132

```
m5=idgrey('patient3',[0;0;0;0],'c');
mp5sleep=pem(m5,zsleepv)
A5=mp5sleep.a;
B5=mp5sleep.b;
C5=mp5sleep.c;
D5=mp5sleep.d;
```
Patient 3 on Patient 7
```
m7=idgrey('patient3',[0;0;0;0],'c');
mp7sleep=pem(m7,zsleepe)
A7=mp7sleep.a;
B7=mp7sleep.b;
C7=mp7sleep.c;
D7=mp7sleep.d;
```
Patient 3 on Patient 2
```
m2=idgrey('patient3',[0;0;0;0],'c');
mp2sleep=pem(m2,zsleepe)
A2=mp2sleep.a;
B2=mp2sleep.b;
C2=mp2sleep.c;
D2=mp2sleep.d;
```
Patient 3 on Patient 6
```
m6=idgrey('patient3',[0;0;0;0],'c');
mp6sleep=pem(m6,zsleepe)
A6=mp6sleep.a;
B6=mp6sleep.b;
C6=mp6sleep.c;
D6=mp6sleep.d;
```
Patient 2 on Patient 1
```
m1=idgrey('patient2',[0;0;0;0],'c');
mp1sleep=pem(m1,zsleepe);
A1=mp1sleep.a;
B1=mp1sleep.b;
C1=mp1sleep.c;
D1=mp1sleep.d;
```
Patient 2 on Patient 5
```
m5=idgrey('patient2',[0;0;0;0],'c');
```

```
mp5sleep=pem(m5,zsleepe)
A5=mp5sleep.a;
B5=mp5sleep.b;
C5=mp5sleep.c;
D5=mp5sleep.d;
```
Patient 2 on Patient 7
```
m7=idgrey('patient2',[0;0;0;0],'c');
mp7sleep=pem(m7,zsleepe)
A7=mp7sleep.a;
B7=mp7sleep.b;
C7=mp7sleep.c;
D7=mp7sleep.d;
```
Patient 2 on Patient 3
```
m3=idgrey('patient2',[0;0;0;0],'c');
mp3sleep=pem(m3,zsleepe)
A3=mp3sleep.a;
B3=mp3sleep.b;
C3=mp3sleep.c;
D3=mp3sleep.d;
```
Patient 2 on Patient 6
```
m6=idgrey('patient2',[0;0;0;0],'c');
mp6sleep=pem(m6,zsleepe)
A6=mp6sleep.a;
B6=mp6sleep.b;
C6=mp6sleep.c;
D6=mp6sleep.d;
```
Patient 1 MISO
```
m1=idgrey('patient1MISO',[0;0;0;1;1;1;1;1;1;1;1;1],'c');
mp1sleep=pem(m1,zsleepe)
A1=mp1sleep.a;
B1=mp1sleep.b;
C1=mp1sleep.c;
D1=mp1sleep.d;
```
Patient 2 MISO
```
m2=idgrey('patient2MISO',[0;0;0;1;1;1;1;1;1;1;1;1],'c');
mp2sleep=pem(m2,zsleepe)
```

134

```matlab
A2=mp2sleep.a;
B2=mp2sleep.b;
C2=mp2sleep.c;
D2=mp2sleep.d;
```
Patient 3 MISO
```matlab
m3=idgrey('patient3MISO',[0;0;0;1;1;1;1;1;1;1;1;1],'c');
mp3sleep=pem(m3,zsleepe)
A3=mp3sleep.a;
B3=mp3sleep.b;
C3=mp3sleep.c;
D3=mp3sleep.d;
```
Patient 5 MISO
```matlab
m5=idgrey('patient5MISO',[0;0;0;1;1;1;1;1;1;1;1;1],'c');
mp5sleep=pem(m5,zsleepe)
A5=mp5sleep.a;
B5=mp5sleep.b;
C5=mp5sleep.c;
D5=mp5sleep.d;
```
Patient 6 MISO
```matlab
m6=idgrey('patient6MISO',[0;0;0;1;1;1;1;1;1;1;1;1],'c');
mp6sleep=pem(m6,zsleepe)
A6=mp6sleep.a;
B6=mp6sleep.b;
C6=mp6sleep.c;
D6=mp6sleep.d;
```
Patient 7 MISO
```matlab
m7=idgrey('patient7MISO',[0;0;0;1;1;1;1;1;1;1;1;1],'c');
mp7sleep=pem(m7,zsleepe)
A7=mp7sleep.a;
B7=mp7sleep.b;
C7=mp7sleep.c;
D7=mp7sleep.d;
```
BP Patient 1 MISO
```matlab
m1=idgrey('patient1MISOBP',[0;0;0],'c');
mp1sleepBP=pem(m1,zsleepe)
A1=mp1sleepBP.a;
```

135

```
B1=mp1sleepBP.b;
C1=mp1sleepBP.c;
D1=mp1sleepBP.d;
```
Patient 2 MISO
```
m2=idgrey('patient2MISOBP',[0;0;0],'c');
mp2sleepBP=pem(m2,zsleepe)
A2=mp2sleepBP.a;
B2=mp2sleepBP.b;
C2=mp2sleepBP.c;
D2=mp2sleepBP.d;
```
Patient 3 MISO
```
m3=idgrey('patient3MISOBP',[0;0;0;0;-1],'c');
mp3sleepBP=pem(m3,zsleepe)
A3=mp3sleepBP.a;
B3=mp3sleepBP.b;
C3=mp3sleepBP.c;
D3=mp3sleepBP.d;
```
Patient 5 MISO
```
m5=idgrey('patient5MISOBP',[0;0;0],'c');
mp5sleepBP=pem(m5,zsleepv)
A5=mp5sleepBP.a;
B5=mp5sleepBP.b;
C5=mp5sleepBP.c;
D5=mp5sleepBP.d;
```
Patient 6 MISO
```
m6=idgrey('patient6MISOBP',[1;1;1;1;1;1],'c');
mp6sleepBP=pem(m6,zsleepe)
A6=mp6sleepBP.a;
B6=mp6sleepBP.b;
C6=mp6sleepBP.c;
D6=mp6sleepBP.d;
```
Patient 7 MISO
```
m7=idgrey('patient7MISOBP',[0,0,0],'c');
mp7sleepBP=pem(m7,zsleepv)
A7=mp7sleepBP.a;
B7=mp7sleepBP.b;
```

136

```
C7=mp7sleepBP.c;
D7=mp7sleepBP.d;
function [A,B,C,D,K,x0] = patient1(par,T,aux)
A = [-.0041519 -.040075 -.031907 -.03907;0.020717 -0.084443 -.15326 -.24221;
-.00056058 -.0082201 .10342 .28553;.0047522 .0094695 -.5456 -.19459]
  +[-0.0046989; 0.02576; 0.00038384; 0.48166]*[par(1) par(2) par(3) par(4)];
B = [-0.0046989; 0.02576; 0.00038384; 0.48166];
C = [136.01 -9.7696 2.3834 -0.97397];
D = zeros(1,1);
K = zeros(4,1);
x0 =[0;0;0;0];
T=5;
end
function [A,B,C,D,K,x0] = patient2(par,T,aux)
A = [-0.0026 -0.0279 -0.0025 -0.0244; 0.0177 -0.0864 0.0056 -0.2098; -0.0008
-0.0104 0.0066 0.1733; 0.0013 -0.0431 -0.1563 -0.1275]
  +[0.0003; 0.0010 ;0.0015;0.0015]*[par(1) par(2) par(3) par(4)];
B = [-0.0046989; 0.02576; 0.00038384; 0.48166];
C = [237.2070 -11.6216 0.5634 -2.9660];
D = zeros(1,1);
K = zeros(4,1);
x0 =[0;0;0;0];
T=5;
end
function [A,B,C,D,K,x0] = patient3(par,T,aux)
A = [-.0010 -.0276 0.0235 .0076;0.0088 -0.0494 .1102 .0505; .0018 .0445 -
.1327 -.3432;.0004 -.0125 .2824 .0256]+[-0.0082; -0.0185; -0.0166; -0.0381]*[par(1)
par(2) par(3) par(4)];
B = [-0.0082; -0.0185; -0.0166; -0.0381];
C = [173.0765 -9.0676 2.5768 -2.4800];
D = zeros(1,1);
K = zeros(4,1);
x0 =[0;0;0;0];
T=5;
end
function [A,B,C,D,K,x0] = patient5(par,T,aux)
```

137

```
A = [-.0046 -.0301 .0327 -.0146;.0070 -.1072 .1349 .1266;-.0054 -.0480 -.0139
.2888;.0058 -.0366 -.1753 -.0239]+[-0.0070; 0.0833; 0.1311; -0.0873]*[par(1)
par(2) par(3) par(4)];
   B = [-0.0070; 0.0833; 0.1311; -0.0873];
   C = [139.8580 -9.8886 0.4684 -0.5230];
   D = zeros(1,1);
   K = zeros(4,1);
   x0 =[0;0;0;0];
   T=5;
   end
   function [A,B,C,D,K,x0] = patient6(par,T,aux)
   A = [-.0055 .0096 -.0862 .0357;.0039 -.0137 -.1087 .1693; .0118 -.0580 -.1745
.0230;.0001 -.0924 .1051 -.0576]+[0.0025; 0.0072; 0.0267; 0.0151]*[par(1) par(2)
par(3) par(4)];
   B = [0.0025; 0.0072; 0.0267; 0.0151];
   C = [137.6774 -8.9751 -5.8180 3.934];
   D = zeros(1,1);
   K = zeros(4,1);
   x0 =[0;0;0;0];
   T=5;
   end
   function [A,B,C,D,K,x0] = patient(par,T,aux)
   A = [-.002193 -.03069 -0.00781 -0.01234;.0165 -.07587 -.07758 -.1197; -.0008685
.01699 -.01573 -.2734;.00157 .02533 .2479 .01591]+[-0.0016; -0.0078; -0.0118;
0.0082]*[par(1) par(2) par(3) par(4)];
   B = [-0.0016; -0.0078; -0.0118; 0.0082];
   C = [175.3789 -10.2103 -0.8136 0.3883];
   D = zeros(1,1);
   K = zeros(4,1);
   x0 =[0;0;0;0];
   T=5;
   end
   function [A,B,C,D,K,x0] = patient1MISO(par,T,aux)
   A = [-.0041519 -.040075 -.031907 -.03907; 0.020717 -0.084443 -.15326 -
.24221; -.00056058 -.0082201 .10342 .28553;.0047522 .0094695 -.5456 -.19459];
```

138

```
B = [-0.0046989 par(1) par(2) par(3); 0.02576 par(4) par(5) par(6); 0.00038384
par(7) par(8) par(9); 0.48166 par(10) par(11) par(12)];
C = [136.01 -9.7696 2.3834 -0.97397];
D = zeros(1,4);
K = zeros(4,1);
x0 =[0;0;0;0];
T=5;
end
function [A,B,C,D,K,x0] = patient2MISO(par,T,aux)
A = [-0.0026 -0.0279 -0.0025 -0.0244; 0.0177 -0.0864 0.0056 -0.2098;-0.0008
-0.0104 0.0066 0.1733;0.0013 -0.0431 -0.1563 -0.1275];
B = [0.0003 par(1) par(2) par(3); 0.0010 par(4) par(5) par(6); 0.0015 par(7)
par(8) par(9); 0.0015 par(10) par(11) par(12)];
C = [237.2070 -11.6216 0.5634 -2.9660];
D = zeros(1,4);
K = zeros(4,1);
x0 =[0;0;0;0];
T=5;
end
function [A,B,C,D,K,x0] = patient3MISO(par,T,aux)
A = [-.0010 -0.0276 0.0235 0.0076; 0.0088 -0.0494 0.1102 .0505;0.0018 0.0445
-0.1327 -0.3432; 0.0004 -0.0125 0.2824 0.0256];
B = [-0.0082 par(1) par(2) par(3); -0.0185 par(4) par(5) par(6); -0.0166 par(7)
par(8) par(9); -0.0381 par(10) par(11) par(12)];
C = [173.0765 -9.0676 2.5768 -2.4800];
D = zeros(1,4);
K = zeros(4,1);
x0 =[0;0;0;0];
T=5;
end
function [A,B,C,D,K,x0] = patient5MISO(par,T,aux)
A = [-0.0046 -0.0301 0.0327 -0.0146; 0.0070 -0.1072 0.1349 .1266; -0.0054
-0.0480 -0.0139 0.2888; 0.0058 -0.0366 -0.1753 -0.0239];
B = [-0.0070 par(1) par(2) par(3); 0.0833 par(4) par(5) par(6); 0.1311 par(7)
par(8) par(9); -0.0873 par(10) par(11) par(12)];
C = [175.3789 -10.2101 -0.8316 0.3883];
```

139

```matlab
D = zeros(1,4);
K = zeros(4,1);
x0 =[0;0;0;0];
T=5;
end
function [A,B,C,D,K,x0] = patient6MISO(par,T,aux)
A = [-0.0055 0.0096 -0.0862 0.0357; 0.0039 -0.0137 -0.1087 0.1693; 0.0118
-0.0580 -0.1745 0.0230; 0.0001 -0.0924 0.1051 -0.0576];
B = [0.0025 par(1) par(2) par(3); 0.0072 par(4) par(5) par(6); 0.0267 par(7)
par(8) par(9); 0.0151 par(10) par(11) par(12)];
C = [137.6774 -8.9751 -5.8180 3.9340];
D = zeros(1,4);
K = zeros(4,1);
x0 =[0;0;0;0];
T=5;
end
function [A,B,C,D,K,x0] = patient7MISO(par,T,aux)
A = [-0.0022 -0.0307 -0.0078 -0.0123; 0.0165 -0.0759 -0.0776 -0.1197;-0.0009
0.0170 -0.0157 -0.2734;0.0016 0.0253 0.2479 0.0159];
B = [-0.0016 par(1) par(2) par(3); -0.0078 par(4) par(5) par(6); -0.0118 par(7)
par(8) par(9); 0.0082 par(10) par(11) par(12)];
C = [175.3789 -10.2101 -0.8316 0.3883];
D = zeros(1,4);
K = zeros(4,1);
x0 =[0;0;0;0];
T=5;
end
```

APPENDIX B

STATE-FEEDBACK MATLAB CODE

L1 Adaptive State Feedback control for Patient 1 Sedated

Reference

```
stepfinal = -15;
```
Reference system parameters
```
Am = Asleep4-Bsleep4*KS5;
Bm = Bsleep4;
Cm=Csleep4;
Dm=Dsleep4;
```
Patient Model A = Asleep4;
```
B=Bsleep4;
C =Csleep4;
D = Dsleep4;
x_0=Csleep4'*inv(Csleep4*Csleep4')*17
```
Adaptive Law Parameters
```
Gamma=1000;
Proj_max_theta=15;
Proj _min_theta=-15;
Theta_max=60;
eps_Theta=.01;
Proj_0 _theta=[0];
Q=2*eye(4);
P=lyap(Am',Q);
Pb=P*B;
```
Control Law
```
k=0.5;
kg=(-1/(C*inv(Am)*B));
```
Tracking Error Stats

```
load inputdata
load reference
load simdata
load trackingerror
rn=(norm(ref(2,:)-BIS(2,:)) ^ 2)/(norm(ref(2,:)) ^ 2)
```

Plot results
```
load inputdata
load reference
load simdata
load stateerror
load sigmadata
figure(1);
subplot(2,1,1);
hold on;
plot((1/60)*BIS(1,:),BIS(2,:),'b');
plot((1/60)*ref(1,:),ref(2,:),'r');
grid on;
title('Patient 1');
xlabel('Time t [min]')
ylabel('BIS')
legend('simBIS','r(t)');
subplot(2,1,2);
plot((1/60)*ISO(1,:),ISO(2,:),'b');
hold on;
grid on;
xlabel('Time t [min]');
ylabel('
```

L1 Adaptive State Feedback control for Patient 2 Sedated

Reference system parameters
```
Am = Asleep4-Bsleep4*KS5;
Bm = Bsleep4;
Cm=Csleep4;
Dm=Dsleep4ISO;
```

142

```matlab
Patient Model A = Asleep4;
B=Bsleep4;
C =Csleep4;
D = Dsleep4ISO;
x_0=Csleep4'*inv(Csleep4*Csleep4')*16.15
Adaptive Law Parameters
Gamma=100000;
Proj_max _theta=1100;
Proj_min _theta=-1100;
Theta_max=1100;
eps_Theta=.01;
Proj_0_theta=0;
Q=2*eye(4);
P=lyap(Am',Q);
Pb=P*B;
Control Law
k=.01;
kg=(-1/(C*inv(Am)*B));
Tracking Error Stats
load inputdata
load reference
load simdata
load trackingerror
rn=(norm(ref(2,:)-BIS(2,:)) ^ 2)/(norm(ref(2,:)) ^ 2)
Plot results
load inputdata
load reference
load simdata
load stateerror
load sigmadata
figure(1);
subplot(2,1,1);
hold on;
plot((1/60)*BIS(1,:),BIS(2,:),'b');
plot((1/60)*ref(1,:),ref(2,:),'r');
grid on;
```

143

```matlab
title('Patient 2');
xlabel('Time t [min]')
ylabel('BIS')
legend('simBIS','r(t)');
subplot(2,1,2);
plot((1/60)*ISO(1,:),ISO(2,:),'b');
hold on;
grid on;
xlabel('Time t [min]');
ylabel('
```

L1 Adaptive State Feedback control for Patient 3 Sedated

Reference system parameters
```matlab
Am = Asleep4-Bsleep4*KS5;
Bm = Bsleep4;
Cm=Csleep4;
Dm=Dsleep4;
```
Patient Model
```matlab
A = Asleep4;
B=Bsleep4;
C =Csleep4;
D = Dsleep4ISO;
x_0=Csleep4'*inv(Csleep4*Csleep4')*16
```
Adaptive Law Parameters
```matlab
Gamma=1000;
Proj_max_theta=20;
Proj_min_theta=-20;
Theta_max=80;
eps_Theta=.01;
Proj_0_theta=[0];
Q=2*eye(4);
P=lyap(Am',Q);
Pb=P*B;
```
Control Law

144

```
k=0.5;
kg=(-1/(C*inv(Am)*B));
```
Tracking Error Stats
```
load inputdata
load reference
load simdata
load stateerror
rn=(norm(ref(2,:)-BIS(2,:)) ^ 2)/(norm(ref(2,:)) ^ 2)
```
Plot results
```
load inputdata
load reference
load simdata
load stateerror
load sigmadata
figure(1);
subplot(2,1,1);
hold on;
plot((1/60)*BIS(1,:),BIS(2,:),'b');
plot((1/60)*ref(1,:),ref(2,:),'r');
grid on;
title('Patient 3');
xlabel('Time t [min]')
ylabel('BIS')
legend('simBIS','r(t)');
subplot(2,1,2);
plot((1/60)*ISO(1,:),ISO(2,:),'b');
hold on;
grid on;
xlabel('Time t [min]');
ylabel('
```

L1 Adaptive State Feedback control for Patient 5 Sedated

Reference system parameters
```
Am = Asleep4-Bsleep4*KS5;
```

145

```
Bm = Bsleep4;
Cm=Csleep4;
Dm=Dsleep4ISO;
```
Patient Model
```
A = Asleep4;
B=Bsleep4;
C =Csleep4;
D = Dsleep4ISO;
x_0=Csleep4'*inv(Csleep4*Csleep4')*17.4
```
Adaptive Law Parameters
```
Gamma=100000;
Proj_max_theta=75;
Proj_min_theta=-75;
Theta_max=1000;
eps_Theta=.01;
Proj_0_theta=[0];
Q=2*eye(4);
P=lyap(Am',Q);
Pb=P*B;
```
Control Law
```
k=1;
kg=(-1/(C*inv(Am)*B));
```
Tracking Error Stats
```
load inputdata
load reference
load simdata
load trackingerror
rn=(norm(ref(2,:)-BIS(2,:)) ^ 2)/(norm(ref(2,:)) ^ 2)
```
Plot results
```
load inputdata
load reference
load simdata
load stateerror
load sigmadata
figure(1);
subplot(2,1,1);
```

146

```
hold on;
plot((1/60)*BIS(1,:),BIS(2,:),'b');
plot((1/60)*ref(1,:),ref(2,:),'r');
grid on;
title('Patient 5');
xlabel('Time t [min]')
ylabel('BIS')
legend('simBIS','r(t)');
subplot(2,1,2);
plot((1/60)*ISO(1,:),ISO(2,:),'b');
hold on;
grid on;
xlabel('Time t [min]');
ylabel('
```

L1 Adaptive State Feedback control for Patient 6 Sedated

Reference system parameters
```
Am = Asleep4-Bsleep4*KS5;
Bm = Bsleep4;
Cm=Csleep4;
Dm=Dsleep4ISO;
```
Patient Model
```
A = Asleep4;
B=Bsleep4;
C =Csleep4;
D = Dsleep4ISO;
x_0=Csleep4'*inv(Csleep4*Csleep4')*10
```
Adaptive Law Parameters
```
Gamma=100000000;
Proj_max_theta=70;
Proj_min_theta=-70;
Theta_max=1000;
eps_Theta=.01;
Proj_0_theta=[0];
```

147

```
Q=2*eye(4);
P=lyap(Am',Q);
Pb=P*B;
```
Control Law
```
k=.1;
kg=(-1/(C*inv(Am)*B));
```
Tracking Error Stats
```
load inputdata
load reference
load simdata
load trackingerror
rn=(norm(ref(2,:)-BIS(2,:))^2)/(norm(ref(2,:))^2)
```
Plot results
```
load inputdata
load reference
load simdata
load stateerror
load sigmadata
figure(1);
subplot(2,1,1);
hold on;
plot((1/60)*BIS(1,:),BIS(2,:),'b');
plot((1/60)*ref(1,:),ref(2,:),'r');
grid on;
title('Patient 6');
xlabel('Time t [min]')
ylabel('BIS')
legend('simBIS','r(t)');
subplot(2,1,2);
plot((1/60)*ISO(1,:),ISO(2,:),'b');
hold on;
grid on;
xlabel('Time t [min]');
ylabel('
```

L1 Adaptive State Feedback control for Patient 7 Sedated

Reference system parameters

```
Am = Asleep4-Bsleep4*KS5;
Bm = Bsleep4;
Cm=Csleep4;
Dm=Dsleep4ISO;
```

Patient Model

```
A = Asleep4;
B=Bsleep4;
C =Csleep4;
D = Dsleep4ISO;
x_0=Csleep4'*inv(Csleep4*Csleep4')*19
```

Adaptive Law Parameters

```
Gamma=1000;
Proj_max_theta=75;
Proj_min_theta=-75;
Theta_max=1000;
eps_Theta=.01;
Proj_0_theta=[0];
Q=2*eye(4);
P=lyap(Am',Q);
Pb=P*B;
```

Control Law

```
k=1;
kg=(-1/(C*inv(Am)*B));
```

Tracking Error Stats

```
load inputdata
load reference
load simdata
load trackingerror
rn=(norm(ref(2,:)-BIS(2,:))^2)/(norm(ref(2,:))^2)
```

Plot results

```
load inputdata
load reference
load simdata
```

149

```
load stateerror
load sigmadata
figure(1);
subplot(2,1,1);
hold on;
plot((1/60)*BIS(1,:),BIS(2,:),'b');
plot((1/60)*ref(1,:),ref(2,:),'r');
grid on;
title('Patient 7');
xlabel('Time t [min]')
ylabel('BIS')
legend('simBIS','r(t)');
subplot(2,1,2);
plot((1/60)*ISO(1,:),ISO(2,:),'b');
hold on;
grid on;
xlabel('Time t [min]');
ylabel('
```

APPENDIX C

OUTPUT-FEEDBACK MATLAB CODE

L1 Adaptive Output Feedback control for Patient 1 Sedated

Reference system parameters

```
num = [1/30];
den = [1 1/30];
```

Patient Model

```
A = Asleep4;
B=Bsleep4;
C =Csleep4;
D = Dsleep4;
x_0=-Csleep4'*inv(Csleep4*Csleep4')*17
```

Adaptive Law Parameters

```
Gamma=50000;
Proj_max_theta=100;
Proj_min_theta=-100;
Theta_max=100;
eps_Theta=.01;
Proj_0_theta=[0];
P=1
```

Control Law

```
omega = 0.001
```

Tracking Error Stats

```
load inputdata
load reference
load simdata
load trackingerror
rn=(norm(ref(2,:)-BIS(2,:))^2)/(norm(ref(2,:))^2)
```

Plot results

```
load inputdata
load reference
load simdata
load stateerror
load sigmadata
load simdata1
figure(1);
subplot(2,1,1);
hold on;
plot((1/60)*BIS(1,:),BIS(2,:),'b');
plot((1/60)*ref(1,:),ref(2,:),'r');
grid on;
title('Patient 1');
xlabel('Time t [min]')
ylabel('BIS')
legend('simBIS','r(t)');
subplot(2,1,2);
plot((1/60)*ISO(1,:),ISO(2,:),'b');
hold on;
grid on;
xlabel('Time t [min]');
ylabel('
```

L1 Adaptive Output Feedback control for Patient 2 Sedated

Reference system parameters
```
num = [1/60];
den = [1 1/60];
```
Patient Model
```
A = Asleep4;
B=Bsleep4;
C =Csleep4;
D = Dsleep4;
x_0=-Csleep4'*inv(Csleep4*Csleep4')*16.15
```
Adaptive Law Parameters

```
Gamma=50000;
Proj_max_theta=100;
Proj_min_theta=-100;
Theta_max=100;
eps_Theta=.01;
Proj_0_theta=[0];
P=1
```

Control Law

```
omega = 0.002
```

Tracking Error Stats

```
load inputdata
load reference
load simdata
load trackingerror
rn=(norm(ref(2,:)-BIS(2,:))^2)/(norm(ref(2,:))^2)
```

Plot results

```
load inputdata
load reference
load simdata
load stateerror
load sigmadata
load simdata1
figure(1);
subplot(2,1,1);
hold on;
plot((1/60)*BIS(1,:),BIS(2,:),'b');
plot((1/60)*ref(1,:),ref(2,:),'r');
grid on;
title('Patient 2');
xlabel('Time t [min]')
ylabel('BIS')
legend('simBIS','r(t)');
subplot(2,1,2);
plot((1/60)*ISO(1,:),ISO(2,:),'b');
hold on;
grid on;
```

153

```
xlabel('Time t [min]');
ylabel('
```

L1 Adaptive Output Feedback control for Patient 3 Sedated

Reference system parameters
```
num = [1/15];
den = [1 1/15];
```
Patient Model
```
A = Asleep4;
B=Bsleep4;
C =Csleep4;
D = Dsleep4;
x_0=-Csleep4'*inv(Csleep4*Csleep4')*16
```
Adaptive Law Parameters
```
Gamma=50000;
Proj_max_theta=100;
Proj_min_theta=-100;
Theta_max=100;
eps_Theta=.01;
Proj_0_theta=[0];
P=1
```
Control Law
```
omega = 0.008
```
Tracking Error Stats
```
load inputdata
load reference
load simdata
load trackingerror
rn=(norm(ref(2,:)-BIS(2,:))^ 2)/(norm(ref(2,:))^ 2)
```
Plot results
```
load inputdata
load reference
load simdata
load stateerror
```

```
load sigmadata
load simdata1
figure(1);
subplot(2,1,1);
hold on;
plot((1/60)*BIS(1,:),BIS(2,:),'b');
plot((1/60)*ref(1,:),ref(2,:),'r');
grid on;
title('Patient 3');
xlabel('Time t [min]')
ylabel('BIS')
legend('simBIS','r(t)');
subplot(2,1,2);
plot((1/60)*ISO(1,:),ISO(2,:),'b');
hold on;
grid on;
xlabel('Time t [min]');
ylabel('
```

L1 Adaptive Output Feedback control for Patient 5 Sedated

Reference system parameters
```
num = [1/30];
den = [1 1/30];
```
Patient Model
```
A = Asleep4;
B=Bsleep4;
C =Csleep4;
D = Dsleep4;
x_0=-Csleep4'*inv(Csleep4*Csleep4')*17.4
```
Adaptive Law Parameters
```
Gamma=50000;
Proj_max_theta=100;
Proj_min_theta=-100;
Theta_max=100;
```

155

```matlab
eps_Theta=.01;
Proj_0_theta=[0];
P=1
```

Control Law

```matlab
omega = 0.002
```

Tracking Error Stats

```matlab
load inputdata
load reference
load simdata
load trackingerror
rn=(norm(ref(2,:)-BIS(2,:))^2)/(norm(ref(2,:))^2)
```

Plot results

```matlab
load inputdata
load reference
load simdata
load stateerror
load sigmadata
load simdata1
figure(1);
subplot(2,1,1);
hold on;
plot((1/60)*BIS(1,:),BIS(2,:),'b');
plot((1/60)*ref(1,:),ref(2,:),'r');
grid on;
title('Patient 5');
xlabel('Time t [min]')
ylabel('BIS')
legend('simBIS','r(t)');
subplot(2,1,2);
plot((1/60)*ISO(1,:),ISO(2,:),'b');
hold on;
grid on;
xlabel('Time t [min]');
ylabel('
```

156

L1 Adaptive Output Feedback control for Patient 6 Sedated

Reference system parameters

```
num = [1/30];
den = [1 1/30];
```

Patient Model

```
A = Asleep4;
B=Bsleep4;
C =Csleep4;
D = Dsleep4;
x_0=-Csleep4'*inv(Csleep4*Csleep4')*10
```

Adaptive Law Parameters

```
Gamma=50000;
Proj_max_theta=100;
Proj_min_theta=-100;
Theta_max=100;
eps_Theta=.01;
Proj_0_theta=[0];
P=1
```

Control Law

```
omega = 0.004
```

Tracking Error Stats

```
load inputdata
load reference
load simdata
load trackingerror
rn=(norm(ref(2,:)-BIS(2,:))^2)/(norm(ref(2,:))^2)
```

Plot results

```
load inputdata
load reference
load simdata
load stateerror
load sigmadata
load simdata1
figure(1);
subplot(2,1,1);
```

```matlab
hold on;
plot((1/60)*BIS(1,:),BIS(2,:),'b');
plot((1/60)*ref(1,:),ref(2,:),'r');
grid on;
title('Patient 6');
xlabel('Time t [min]')
ylabel('BIS')
legend('simBIS','r(t)');
subplot(2,1,2);
plot((1/60)*ISO(1,:),ISO(2,:),'b');
hold on;
grid on;
xlabel('Time t [min]');
ylabel('
```

L1 Adaptive Output Feedback control for Patient 7 Sedated

Reference system parameters

```matlab
num = [1/60];
den = [1 1/60];
```

Patient Model

```matlab
A = Asleep4;
B=Bsleep4;
C =Csleep4;
D = Dsleep4;
x_0=-Csleep4'*inv(Csleep4*Csleep4')*10
```

Adaptive Law Parameters

```matlab
Gamma=50000;
Proj_max_theta=100;
Proj_min_theta=-100;
Theta_max=100;
eps_Theta=.01;
Proj_0_theta=[0];
P=1
```

Control Law

158

omega = 0.004

Tracking Error Stats

load inputdata

load reference

load simdata

load trackingerror

rn=(norm(ref(2,:)-BIS(2,:))$\,\hat{}\,$2)/(norm(ref(2,:))$\,\hat{}\,$2)

Plot results

load inputdata

load reference

load simdata

load stateerror

load sigmadata

load simdata1

figure(1);

subplot(2,1,1);

hold on;

plot((1/60)*BIS(1,:),BIS(2,:),'b');

plot((1/60)*ref(1,:),ref(2,:),'r');

grid on;

title('Patient 7');

xlabel('Time t [min]')

ylabel('BIS')

legend('simBIS','r(t)');

subplot(2,1,2);

plot((1/60)*ISO(1,:),ISO(2,:),'b');

hold on;

grid on;

xlabel('Time t [min]');

ylabel('

159

Checking stability of H(s)

Patient 1

ParaM=1/30;
ParaOmega=.008;
poly=ParaOmega*([0 1.359 1.362 -0.08081 0.00635 0]+ParaM*[0 0 1.359 1.362 -0.08081 0.00635])+ParaM*[1 .1798 .1461 .01316 .0001649 0];
roots(poly)

Patient 2

ParaM=1/60;
ParaOmega=.0001;
poly=ParaOmega*([0 -0.05594 -0.001404 -0.002921 0.0002398 0]+ParaM*[0 0 -0.05594 -0.001404 -0.002921 0.0002398])+ParaM*[1 .1798 .1461 .01316 .0001649 0];
roots(poly)

Patient 3

ParaM=1/60;
ParaOmega=.0001;
poly=ParaOmega*([0 1.2 0.1397 0.06413 0.0002461 0]+ParaM*[0 0 1.2 0.1397 0.06413 0.0002461])+ParaM*[1 .1575 .09489 .003838 2.527e-05 0];
roots(poly)

Patient 5

ParaM=1/15;
ParaOmega=.005;
poly=ParaOmega*([0 2.221 -0.2494 0.09021 0.002341 0]+ParaM*[0 0 2.221 -0.2494 0.09021 0.002341])+ParaM*[1 .1496 .06726 0.006355 5.131e-05 0];
roots(poly)

160

Patient 6

ParaM=1/30;
  ParaOmega=.02;
  poly=ParaOmega*([0 -0.1836 0.1526 -0.003364 0.0005311 0]+ParaM*[0 0 -0.1836 0.1526 -0.003364 0.0005311])+ParaM*[1 0.2513 0.02248 0.003408 2.944e-05 0];
  roots(poly)

Patient 7

ParaM=1/15;
  ParaOmega=.01;
  poly=ParaOmega*([0 0.188 -0.02198 0.0101 0.0005796 0]+ParaM*[0 0 0.188 -0.02198 0.0101 0.0005796])+ParaM*[1 0.0779 0.07254 0.005276 4.678e-05 0];
roots(poly)

161

APPENDIX D

MATHEMATICAL BACKGROUND

D.1 $\mathscr{L}_1$ norm

The $\mathscr{L}_1$ gain of a BIBO stable proper SISO system is defined as

$$||H(s)||_{\mathscr{L}_1} = \int_0^\infty |h(t)|\,\mathrm{d}t \tag{D.1}$$

where $h(t)$ is the impulse response of $H(s)$.

For a stable, proper m input by n output system, $H(s)$, the $\mathscr{L}_1$-norm of $H(s)$ is given to be

$$||H(s)||_{\mathscr{L}_1} = \max_{i=1,...,n}\left(\sum_{j=1}^{m}||H_{ij}(s)||_{\mathscr{L}_1}\right) \tag{D.2}$$

where $H_{ij}(s)$ is the ith row, j^{th} column element of $H(s)$.

D.2   Projection Operator

The projection operator ensures that the parameter estimate $\hat{\theta}(t)$ remains inside the compact set Θ.

To use the projection operator, we consider a convex compact set with a smooth boundary given by:

$$\Theta_c \triangleq \{\theta \in \mathbb{R}^n | f(\theta) \le c\},\ 0 \le c \le 1, \tag{D.3}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is the following convex function:

$$f(\theta) = \frac{\theta^T \theta - \theta_{max}^2}{\varepsilon_\theta \theta_{max}^2} \tag{D.4}$$

162

where θ_{max} is a bound placed on the norm of the parameter θ and ε_θ is the convergence tolerance. We let the true value of θ, given as $\theta*$, be in the set Θ_0. The projection operator is given by

$$\text{Proj}(\theta,y) \triangleq \begin{cases} y & \text{if } f(\theta) < 0, \\ y & \text{if } f(\theta) \geq 0 \text{ and } \nabla f^T y \leq 0 \\ y - \frac{\nabla f}{||\nabla f||} \left\langle \frac{\nabla f^T}{||\nabla f||}, y \right\rangle f(\theta) & \text{if } f(\theta) \geq 0 \text{ and } \nabla f^T y > 0. \end{cases} \quad \text{(D.5)}$$

The projection operator in equation (D.5) does not change y if θ is in the set $\Theta_0 = \{\theta \in \mathbb{R}^n | f(\theta) \leq 0\}$. Whenever θ is in the set $0 \leq f(\theta) \leq 1$, if $\nabla f^T y > 0$, the projection operator $\text{Proj}(\theta,y)$ subtracts a vector normal to the boundary of $\Theta_c = \{\theta \in \mathbb{R}^n | f(\theta) = c\}$ resulting in a smooth change from the original y vector field to a tangent or inward vector field for $c = 1$. From this, it can be seen that whenever $\dot\theta(t) = \text{Proj}(\theta(t), y(t))$, then θ never leaves Θ_1.

REFERENCES

[1] P. S. Glass, M. Bloom, L. Kearse, C. Rosow, P. Sebel, and P. Manberg, "Bispectral analysis measures sedation and memory effects of propofol, midazolam, isoflurane, and alfentanil in healthy volunteers," *Anesthesiology*, vol. 86, no. 4, pp. 836–847, April 1997.

[2] P. S. Sebel, "Bispectral monitoring technology: Clinical applications," in *Anesthesia for the New Millennium - Modern Anesthetic Clinical Pharmacology*, T. Stanley and T. Egan, Eds. Kluwer Academic Publishers, 1999, pp. 99–104.

[3] B. Guignard, "Monitoring analgesia," *Best Practice & Research Clinical Anaesthesiology*, vol. 20, no. 1, pp. 161–180, 2006.

[4] I. Kalli, "Neuromuscular block monitoring," *Neuromuscular Monitoring in Anesthesia*, no. 13, pp. 442–451, 2003.

[5] A. Sardesai and R. Griffiths, "Monitoring techniques: neuromuscular blockade," *Anesthesia & intensive care medicine*, vol. 6, no. 6, pp. 198–200, 2005.

[6] P. Sebel, E. Lang, I. J. Rampil, P. F. White, R. C. an d M. Jopling, N. T. Smith, P. S. Glass, and P. Manberg, "A multicenter study for bispectral electroencephalogram analysis for monitoring anesthetic effect," *Anesthesia and Analgesia*, pp. 891–899, 1997.

[7] N. H. G. Holford and L. B. Sheiner, "Understanding the dose-effect relationship: Clinical application of pharmacokinetic-pharmacodynamic models," *Clinical Pharmacokinetics*, vol. 6, pp. 429–453, 1981.

[8] T. W. Schnider, C. F. Minto, and D. R. Stanski, "The effect compartment concept in pharmacodynamic modelling," *Anesthetic Pharmacology Review*, vol. 2, no. 3, pp. 204–213, 1994.

[9] K. Olkkola and H. Schwilden, "Adaptive closed-loop feedback control of vecuronium-induced neuromuscalar relaxation," *European Journal of Anaesthesiology*, vol. 8, no. 1, pp. 7–12, Jan. 1991.

[10] D. R. Wada and D. S. Ward, "The hybrid model: A new pharmacokinetic model for computer-controlled infusion pumps," *IEEE Trans. on Biomed. Eng.*, vol. 41, no. 2, pp. 134–142, Feb. 1994.

[11] J. R. Jacobs and E. A. Williams, "Algorithm to control effect compartment drug concentrations in pharmocokinetic model-driven drug delivery," *IEEE Trans. on Biomed. Eng.*, vol. 40, no. 10, pp. 993–999, Oct. 1993.

[12] H. H. Lin, C. L. Beck, and M. J. Bloom, "On the use of multivariable piecewise-linear models for predicting human response to anesthesia," *IEEE Trans. on Biomedical Engineering*, vol. 51, no. 11, pp. 1876–1887, 2004.

[13] C. L. Beck, H.-H. Lin, and M. Bloom, "Modeling and control of anesthetic pharmacodynamics," in *Biology and Control Theory: Current Challenges*, S. Tarbouriech, I. Queinnec, G. Garcia, and S. Niculescu, Eds. Springer-Verlag Lecture Notes Series, 2007, vol. 357.

[14] C. Cao and N. Hovakimyan, "Design and analysis of a novel L_1 adaptive control architeture with guaranteed transient performance," *IEEE Trans. on Auto. Control*, vol. 53, no. 2, pp. 586–591, 2008.

[15] C. Cao and N. Hovakimyan, "L_1 adaptive output feedback controller for systems of unknown dimension," in *IEEE Trans on Auto. Control*, vol. 53, 2008.

[16] H. Schwilden, H. Stoeckel, and J. Schüttler, "Closed-loop feedback control of propofol anaesthesia by quantitative EEG analysis in humans," *British Journal of Anaesthesia*, pp. 290–296, 1989.

[17] H. Schwilden and H. Stoeckel, "Effective therapeutic infusions produced by closed-loop feedback control of methohexital administration during total intravenous anesthesia with fentanyl," *Anesthesiology*, vol. 73, no. 2, pp. 225–229, Aug. 1990.

[18] H. Schwilden and H. Stoeckel, "Closed-loop feedback controlled administration of alfentanil during alfentanil-nitrous oxide anaesthesia," *British Journal of Anaesthesia*, pp. 389–393, 1993.

[19] C. W. Frei, A. Gentilini, M. Derighetti, A. H. Glattfelder, M. Morari, T. W. Schnider, and A. M. Zbinden, "Automation in anesthesia," in *Proc. American Control Conference*, 1999, pp. 1258–1263.

[20] C. W. Frei, M. Derighetti, M. Morari, A. H. Glattfelder, and A. M. Zbinden, "Improving regulation of mean arterial blood pressure during anesthesia through estimates of surgery effects," *IEEE Trans. on Biomed. Eng.*, vol. 47, no. 11, pp. 1456–1464, Nov. 2000.

[21] A. Gentilini, C. W. Frei, A. H. Glattfelder, M. Morari, T. J. Sieber, R. Wymann, T. W. Schnider, and A. M. Zbinden, "Multi-tasked closed loop control in anesthesia," *IEEE Eng. in Med. and Bio. Magazine*, vol. 20, no. 1, pp. 39–53, Jan. 2001.

165

[22] A. Gentilini, M. Rossoni-Gerosa, C. Frei, R. Wymann, M. Morari, A. Zbinden, and T. Schnider, "Modeling and closed loop control of hypnosis by means of bispectral index (BIS) with isoflurane," *IEEE Trans. on Biomed. Eng.*, vol. 48, no. 8, pp. 874–889, Aug. 2001.

[23] A. Gentilini, C. Schaniel, M. Morari, C. Bieniok, R. Wymann, and T. Schnider, "A new paradigm for the closed-loop intraoperative administration of analgesics in humans," *IEEE Trans. on Biomed. Eng.*, vol. 49, no. 4, pp. 289–299, April 2002.

[24] E. Mortier, M. Struys, T. D. Smet, L. Versichelen, and G. Rolly, "Closed-loop controlled administration of propofol using bispectral analysis," *Anesthesia*, vol. 53, no. 8, pp. 749–754, 1998.

[25] W. M. Haddad, T. Hayakawa, , and J. M. Bailey, "Adaptive control for nonnegative and compartmental dynamical systems with applications to general anesthesia," *Int. J. Adapt. Control Signal Process*, vol. 17, pp. 209–235, 2003.

[26] T. Hayakawa, W. M. Haddad, N. Hovakimyan, and V. Chellaboina, "Neural network adaptive control for nonlinear nonnegative dynamical systems," *IEEE Trans. Neural Networks*, vol. 16, pp. 399–413, 2005.

[27] C. Cao and N. Hovakimyan, "Stability margins of L_1 adaptive controller: Part ii," in *ACC*, 2007, pp. 3931–3936.

[28] V. Patel, C. Cao, N. Hovakimyan, K. Wise, and E. Lavretsky, "L1 adaptive controller for tailless unstable aircraft in the presence of unknown actuator failures," *International Journal of Control*, pp. 705–720, 2009.

[29] J. Wang, V. Patel, C. Cao, N. Hovakimyan, and E. Lavretsky, "Novel L_1 adaptive control methodology for aerial refueling with guaranteed transient performance," *AIAA Journal of Guidance, Control and Dynamics*, pp. 182–193, 2008.

[30] V. V. Chalam, *Adaptive Control Systems: Techniques and Applications*. Marcek Dekker, Inc., 1987.

[31] K. Wise, E. Lavretsky, N. Hovakimyan, and J. Wang, "Verifiable adaptive flight control: Unmanned combat aerial vehicles and aerial refueling," in *AIAA Guidance, Navigation, and Control Conference*, 2008.

[32] C. Cao, V. V. Patel, K. Reddy, N. Hovakimyan, and E. Lavretsky, "Are the phase and time-delay margins always adversely affected by high-gain?" in *Proc. of AIAA Guidance, Navigation and Control Conference*, AIAA-2006-6347.

[33] H. H. Lin, C. L. Beck, and M. Bloom, "Multivariable LPV control of anesthesia delivery during surgery," in *ACC*, 2008, pp. 825–831.

[34] R. Berne, M. N. Levy, B. M. Koeppen, and B. A. Stanton, *Physiology*, 4th ed. Mosby, Inc., St. Louis, MI, 1998.

[35] M. von Spreckelsen and B. Bromm, "Estimation of single-evoked cerebral potentials by means of parametric modeling and kalman filtering," *IEEE Trans. on Biomed. Eng.*, vol. 35, no. 9, pp. 691–700, Sept. 1988.

[36] M. Derighetti, C. W. Frei, M. Buob, A. M. Zbinden, and T. W. Schnider, "Modeling the effect of surgical stimulation on mean arterial blood pressure," in *Proceedings of the 1997 19th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 5, 1997, pp. 2172–2175.

[37] P. Van Overschee and B. De Moor, "N4SID: Subspace algorithms for identification of combined deterministic-stochastic systems," *Automatica*, vol. 30, no. 1, pp. 75–93, 1994.

[38] E. Xargay, N. Hovakimyan, and C. Cao, "L_1 adaptive controller for multi-input multi-output systems in the presence of uncertain nonlinear cross-coupling," in *Proceedings, ACC, pgs 874-879*, 2010.

[39] J. Abenstein and M. Warner, "Anesthesia providers, patient outcomes, and costs," *Anesth. Analg.*, vol. 82, pp. 1273–1283, 1996.

[40] A. Kurz and D. Sessler, "Opioid-induced bowel dysfunction: pathophysiology and potential new therapies," *Drugs*, vol. 63, no. 7, pp. 649–671, 2003.

[41] B. Kabon and A. Kurz, "Optimal perioperative oxygen administration," *Current Opinion in Anesth.*, vol. 19, pp. 11–18, 2006.

[42] H. H. Lin, "Multivariable modeling and control of the response to anesthesia," Ph.D. dissertation, University of Illinois at Urbana-Champaign, May 2006.