ABSTRACT

Title of Thesis:	PREDICTION AND CLOSED-LOOP CONTROL OF BLOOD PRESSURE FOR HEMORRHAGE RESUSCITATION
	Drew Xavier Hohenhaus, Master of Science, 2023
Thesis Directed By:	Associate Professor Jin-Oh Hahn Department of Mechanical Engineering

Hemorrhage is responsible for a large percentage of mortality worldwide and the majority of fatalities on the battlefield. Resuscitation procedures for hemorrhage trauma patients are critical for their recovery. Currently, during resuscitation, physicians manually monitor blood pressure and use intuition to determine when fluid should be administered and how much. Due to factors such as exhaustion, distraction, and inexperience of the physician, this method has often been reported as fallible. This thesis proposes two methods to assist in automating hemorrhage resuscitation. The first is a blood pressure prediction algorithm for decision support systems. The algorithm individualizes itself to different subjects using extended Kalman filtering (EKF), to account for high inter-subject variability, before accurately forecasting future blood pressure. The second method is an observer-based feedback controller which regulates blood pressure from a hypotensive state back to a "healthy" setpoint. The controller was designed using linear matrix inequality (LMI) techniques to ensure it was absolutely stable, which let a portion of the hemodynamic plant model remain unspecified and allowed for performance over a range of physiologies. Both strategies were evaluated in-silico on a cohort of 100 virtual patients generated from an experimental dataset. The prediction algorithm showed accuracy superior to conventional

assumptions. The controller tracked the given setpoint with an accuracy and performance comparable to more complex adaptive methods. Further work, with respect to the prediction algorithm, includes developing it into a full decision-support system and incorporating disturbance rejecting components to account for common issues such as rebleed. The controller's performance deteriorates for high-speed applications, suggesting further study is required to increase its situational flexibility.

PREDICTION AND CLOSED-LOOP CONTROL OF BLOOD PRESSURE FOR HEMORRHAGE RESUSCITATION

by

Drew Xavier Hohenhaus

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park, in partial fulfillment
of the requirements for the degree of
Master of Science
2023

Advisory Committee:
Professor Jin-Oh Hahn, Chair
Professor Hosam Fathy
Professor Yancy Diaz-Mercado

© Copyright by
Drew Xavier Hohenhaus
2023

Acknowledgements

To start, I would like to thank my advisor, Dr. Jin-Oh Hahn, who has done a wonderful job guiding me through this process. Despite my many mistakes, miscalculations, and missteps, he has shown immense patience and kindness to always steer me back on course. None of this would have been possible without his help.

I would also like to thank Dr. Ali Tivay, who graciously took the time to explain his work to me and how it pertained to mine. No matter how many times I had to say "Um, I have no idea what you are talking about," he always smiled, nodded, and explained the topic in a different way until I could understand.

Which leads me to the rest of the students I was lucky enough to work with here at UMD: Weidi Yin, Yekanth Chalumuri, Yi-Ming Kao, Jesse Parreira, Yaunyaun Zhou, Parham Rezaei, Sina Masoumi, Azin Mousavi, and Ghazal Arabi Darrehdor. Their support and advice have been instrumental to my journey.

Lastly, I must acknowledge my family. Without my mother, Linda, and my father, Andrew, I would literally not be here. I suppose without their unconditional love and support I would not be here figuratively either. And to my sister, Callie, thank you for being my friend through all of this, and for being always willing to laugh at the absurdity of higher education with me.

Table of Contents

Acknowledgements ii
Table of Contents iii
List of Tables v
List of Figures vi
List of Abbreviations vii
Chapter 1: Introduction 1
1.1 Background 1
1.2 Motivation 1
1.3 Objectives 3
1.4 Outline 4
Chapter 2: Cardiovascular Dynamics Modeling 6
2.1 Comprehensive Cardiovascular Model 6
2.2 Generative Model for Virtual Patients 8
2.3 Hemorrhage Model 10
2.4 Simple Lumped Model 12
Chapter 3: Blood Pressure Prediction Algorithm 14
3.1 Decision Support for Hemorrhage Resuscitation 14
3.2 Predicting Blood Pressure 16
3.2.1 Prediction Model 16
3.2.2 The Extended Kalman Filter 16
3.2.3 Multi-Stage Estimation 18
3.2.4 Forecasting Blood Pressure 20
Chapter 4: Evaluating the Blood Pressure Prediction Algorithm 22
4.1 Simulation Scenario 22
4.2 No Disturbance Performance 23
4.2.1 Single Virtual Patient 23
4.2.2 Virtual Patient Cohort 26
4.3 Rebleed Performance 28
4.3.1 Single Virtual Patient 28
4.3.2 Virtual Patient Cohort 30
4.4 Discussion 31
Chapter 5: Absolutely Stable Closed-Loop Control 33
5.1 Closed-Loop Control for Hemorrhage Resuscitation 33
5.2 Absolute Stability 34
5.3 Observer-Based Integral Control 36
5.4 Design via Linear Matrix Inequalities 39
5.5 Absolutely Stable Observer-Based Integral Controller 42
Chapter 6: Evaluating the Closed-Loop Controller 44
6.1 Simulation Scenario 44
6.2 Controller Performance 45
6.3 Stability Analysis 47
6.4 Discussion 48
Chapter 7: Conclusion and Future Work 51
7.1 Conclusions for the Prediction Algorithm 51

7.2 Conclusions for the Absolutely Stable Controller 52
References 54

List of Tables

Table 2-1: CC model parameters for subject generator model……9

Table 2-2: Hemorrhage and decompensation distributions…….....12

Table 3-1: Summary of BPPA algorithm as implemented before initial infusion.……21

Table 3-2: Summary of BPPA algorithm as implemented after initial infusion……21

Table 5-1: Summary of values used to design controllers…….....43

Table 6-1: Performance metric distributions for both controllers…….....46

List of Figures

Figure 2-1: Example MAP response, showing hemorrhage and rebleed behavior……11

Figure 3-1: Diagram illustrating blood pressure prediction protocol……..……..15

Figure 4-1: EKF estimates for a representative virtual patient after infusion……24

Figure 4-2: MAP predictions for a representative virtual patient.…….25

Figure 4-3: Error progression over prediction period for all 3 prediction lengths.……26

Figure 4-4: Distributions of normalized RMSE for BPPA across the prediction period……...……27

Figure 4-5: EKF estimates, while internal bleeding is occurring, for a representative virtual patient over stage II……...……29

Figure 4-6: MAP predictions for a representative virtual patient, during rebleeding.……30

Figure 4-7: Error progression over prediction period for all 3 prediction lengths, during rebleeding…….……30

Figure 4-8: Distributions of normalized RMSE for BPPA across the prediction period, when rebleeding is occurring……31

Figure 5-1: System compatible with absolute stability analysis……..…34

Figure 5-2: Block diagram of closed-loop system including the SL model plant and observer-based integral controller……..38

Figure 6-1: MAP responses of a representative virtual patient……45

Figure 6-2: MAP responses for all 100 virtual patients…….46

Figure 6-3: CC model's BV to BP ratio relative to sector bounds…….……47

Figure 6-4: Color maps of observer gains and what percentage of SL model virtual patients were absolutely stable…….……48

List of Abbreviations

BP	Blood Pressure
BPPA	Blood Pressure Prediction Algorithm
CC	Comprehensive Cardiovascular
CO	Cardiac Output
CVP	Central Venous Pressure
DCR	Damage Control Resuscitation
DSS	Decision Support System(s)
EKF	Extended Kalman Filter
HCT	Hematocrit
KF	Kalman Filter
LMI	Linear Matrix Inequality
MAP	Mean Arterial Pressure
MDAPE	Median Absolute Performance Error
MDPE	Median Performance Error
MMAE	Multiple Model Adaptive Estimation
RMSE	Root Mean Squared Error
SL	Simple Lumped
SVR	Systemic Vascular Resistance

Chapter 1: Introduction

1.1 Background

The effective delivery of therapies is critical to clinical outcomes, and can be the difference between life and death for seriously ill patients. In today's clinical practices, medication delivery is managed by human clinicians who empirically adjust dosages. As data collection becomes more accurate in clinical spaces, therapies have the potential to be automated, in the hopes of preventing suboptimal treatment. Work on automated therapeutic systems started at least as far back as the 1980s, when closed-loop controllers were developed to administer vasodilators to correct blood pressure (BP) [1]. In recent decades, algorithms to automate the control of insulin, the control of anesthetics, the control of coagulation, and the control of many other therapies have been developed [2]-[4]. However, there remain many challenges in therapy delivery to be addressed, particularly in critical care systems.

1.2 Motivation

Hemorrhagic shock accounts for approximately 40% of trauma related mortality worldwide and over 85% of deaths on the battlefield [5]. The early administration of resuscitation procedures for such trauma patients is critical, with approximately 25% of mortalities preventable with intervention [6]. Despite the extreme demand for effective treatment, current treatment strategies administered by physicians have often been reported as fallible. Factors such as the high variability

among individuals' physiologies and the distinct possibility of unknown disturbances in trauma scenarios often cause current "one-size-fits-all" strategies to struggle [7].

As a consequence, current approaches to treating hemorrhagic shock remain varied and suboptimal. Controversy exists over the timing and type of treatment, in particular. All resuscitation strategies consist of arresting hemorrhage, restoring blood volume, and correcting coagulopathy, acidosis, and hypothermia [8]. The restoration of blood volume typically entails the infusion of fluid to reach a target blood pressure. Currently, physicians manually monitor blood pressure and use intuition to determine when fluid is given. This method is time consuming and prone to human error, as physicians are inevitably distracted, exhausted, or inexperienced [9].

For example, damage control resuscitation (DCR) is a treatment strategy that has become popular from military operations in Iraq and Afghanistan [10]. DCR differs from other resuscitation procedures in that it attempts an earlier and more aggressive correction of coagulopathy and metabolic derangement. One way it does this is through permissive hypotension. Permissive hypotension is the notion that the initial goal of restoring blood volume is to keep the blood pressure low enough to avoid exacerbating uncontrolled hemorrhage while high enough to maintain the perfusion of end organs [10]. This can be particularly difficult as the region between keeping organ perfusion and avoiding further blood loss can be not only uncertain, but very small [11][12]. Optimal permissive hypotension demands a high level of precision that is currently unavailable.

There is clearly the potential and, as permissive hypotension illustrates, the need to automate fluid resuscitation using blood pressure feedback. Computerized

automation would enable precise volume administration to hit blood pressure targets, while relieving physicians to focus on other complications associated with trauma.

1.3 Objectives

This work aims to begin addressing the faults in current resuscitation treatment strategies, by proposing two separate algorithms whose objectives are to automate fluid resuscitation.

The first method is a blood pressure prediction algorithm (BPPA) for decision support systems (DSS). A DSS which predicts blood pressure into the future and recommends the timing and size of upcoming boluses to physicians is envisioned. The system would adapt to individual patients based only on blood pressure measurements, letting physicians achieve a higher level of accuracy in hitting target blood pressures than if they used purely intuition and/or set "one-size-fits-all" strategies. The first step in developing a DSS that can meet these goals is achieving accurate blood pressure forecasting. The proposed BPPA accomplishes this aim by individualizing itself to subjects via a multi-stage nonlinear filtering technique. Extended Kalman filters estimate the parameters of a blood volume kinetics model over multiple stages in a pre-prediction period using blood pressure data, before the model is simulated into the future to make an accurate prediction.

The second approach to automating fluid administration is an observer-based integral controller designed to be absolutely stable using a linear matrix inequality (LMI) technique. In hospital settings, where certain hardware is available, closed-loop controllers can be used to computerize resuscitation procedures to the point a physician is no longer required to be a part of the feedback loop. Prior work has been reported on

closed-loop algorithms for hemorrhage resuscitation, and while these methods were shown to be effective in some cases, they also had drawbacks. Most existing control algorithms come from empirical analysis and lack any analytical foundation for robustness [13]-[14]. Complex adaptive controllers have been developed in response, but require set pre-control procedures [15]. The observer-based integral control proposed in this work finds a middle ground. It is simpler and more flexible to trauma settings than adaptive methods, and yet was designed using LMIs to guarantee stability over highly uncertain systems.

1.4 Outline

This work is organized such that any pieces relevant to both methodologies are detailed first, followed by discussion of each of the individual algorithms.

Chapter 1 contains an introduction to the problem space and proposed treatment strategies of this thesis.

Chapter 2 covers several models of cardiovascular dynamics which are leveraged for both treatment strategies. These include a comprehensive model used for evaluating the algorithms in-silico, and a simplified model used to design the algorithms themselves.

Chapter 3 explains the development of the BPPA. It includes more information on the envisioned decision support system and the details of how the nonlinear filtering is used to individualize the prediction model.

Chapter 4 shows the results of evaluating the BPPA via simulations. It includes discussion on the successes and limitations of the BPPA.

Chapter 5 encompasses the closed-loop controller design. It discusses the challenges associated with developing a controller for uncertain physiological systems and how absolutely stable control was used to address them.

Chapter 6 assesses the performance of the controller in-silico. It also contains a discussion on the controller's relationship between speed and performance, and to what extent absolute stability was necessary.

Chapter 7 summarizes the contributions and suggests high-yield future work to be conducted.

Chapter 2: Cardiovascular Dynamics Modeling

In this chapter, two different models, both describing cardiovascular dynamics, are detailed. The first is an interpretable physics-based model designed to reproduce the effects of major blood loss and infusion in a patient. This model has been fit to animal data in prior work and serves as the basis for in-silico testing. It is referred to as the Comprehensive Cardiovascular (CC) model [16]. The second model is a simplified version of the CC model, designed to facilitate the prediction and control of blood pressure. This model is used within both the prediction algorithm and the close loop controller. It is referred to as the Simple Lumped (SL) model [17].

2.1 Comprehensive Cardiovascular Model

The CC model has three states which account for the blood volume in the arterial and venous spaces along with the red blood cell volume. The blood circulation in a subject is described using these states as follows:

$$\dot{v}_a = Q - \frac{P_a - P_v}{R} - J_H - J_F \tag{2.1}$$

$$\dot{v}_v = -Q + \frac{P_a - P_v}{R} + J_I \tag{2.2}$$

$$\dot{v}_r = -J_H H \tag{2.3}$$

where v_a & v_v are the arterial and venous blood volumes with v_{a0} & v_{v0} as initial values; P_a & P_v are the Mean Arterial Pressure (MAP) and the Central Venous Pressure (CVP) respectively; Q denotes the Cardiac Output (CO), which is the flow-rate of blood pumped by the heart; R indicates the Systemic Vascular Resistance (SVR); J_H & J_I are

the hemorrhage and infusion rates; J_F is the rate of fluid exchanged from the tissue space; v_r is the total red blood cell volume; and H is the blood hematocrit (HCT).

The relationship between the change in blood pressure from a baseline is related to the blood volume as follows:

$$P_a = P_{a0} + K_a\,(v_a - v_{ao}) \tag{2.4}$$

$$P_v = P_{v0} + K_v\,(v_v - v_{v0}) \tag{2.5}$$

where P_{a0} & P_{v0} are the healthy baselines of the MAP and the CVP respectively, and K_a & K_v are the arterial and venous elastance respectively.

An important aspect of the body is its ability to move interstitial fluid from tissue into the blood in reaction to hemorrhage. This exchange is modeled as a feedback controller dependent on the fluid lost or given to the body:

$$J_F = K_F\big(v_a - v_{a_0} + v_v - v_{v_0} - r_F\big) \tag{2.6}$$

$$\dot{r_F} = \frac{1}{1+\alpha_I} J_I - \frac{1}{1+\alpha_H} J_H \tag{2.7}$$

where r_F is the reference signal of the body's internal controller, K_F is the gain of the controller, and α_I & α_H are the fractions of fluid that remain in the tissue space after infusion and hemorrhage.

The cardiac output is the flow rate of blood pumped by the heart. Changes occur in cardiac output due to shifts in CVP affecting the right atrial pressure and due to the body's natural control mechanisms. The cardiac output can be expressed as:

$$Q = Q_0 + \beta_v(P_v - P_{v0}) + Ds_q \tag{2.8}$$

$$\dot{s_q} = -K_Q(Q - Q_0) \tag{2.9}$$

where Q_0 is a baseline cardiac output, K_Q is the sensitivity of cardiac output to changes in preload, β_v represents the sensitivity to venous pressure changes, and s_q is a state variable representing the amount of change in CO directed by the body. D is a decompensation factor that is unique to this work. The decompensation factor prevents the body from correcting its CO. Without it the CC model will eventually return to its baseline values. It serves as a catch-all for complications associated with trauma (e.g. head injury) that would make patients particularly sick [18].

The SVR is resistance of the blood vessels to the flow of blood. The body controls SVR such that satisfactory MAP is maintained for tissue perfusion. SVR is expressed as:

$$R = R_0 + K_h(H - H_0) + Ds_R \tag{2.10}$$

$$\dot{s_R} = -\frac{1}{\tau_R} s_R - \frac{K_R}{\tau_R}\,(P_a - P_{a0}) \tag{2.11}$$

where R_0 is a baseline resistance, K_h is the sensitivity to HCT changes, K_R is the controller gain, τ_R is the controller time constant, and s_R is a state variable for the management of SVR.

2.2 Generative Model for Virtual Patients

In this work, the developed algorithms explained in later sections are evaluated on virtual patients whose physiological characteristics are represented by unique set of CC model parameters. The generative model used to produce these virtual patients was found via a collective variational inferencing methodology.

The generative model was developed as detailed in Tivay et. al. [16]. In short, a hierarchical probabilistic model was built. At the highest level a subject generator was delegated to generate virtual subjects. An intermediate level consisted of applying inputs to each generated subject prompting physiological processes. The last level defined what physiological variables were measured and the associated measurement noise. Variational inferencing was used to fit the probabilistic model to MAP data collected from sheep (N=23) which had undergone controlled hemorrhage and infusion.

The subject generator model is shown as:

$$\theta_i \sim N(\phi_u, \phi_\Sigma) \tag{2.12}$$

where ϕ_u is the mean vector of the generator, ϕ_Σ is the covariance matrix of the generator, and θ_i is a generated subject-specific parameter vector for the cardiovascular dynamics model. θ_i can be written as:

$$\theta_i = \left[v_o \;\; H_0 \;\; Q_0 \;\; P_{a0} \;\; K_v \;\; \frac{K_a}{K_v} \;\; K_F \;\; \alpha_I \;\; \alpha_H \;\; K_H \;\; \tau_R \;\; K_R \;\; \beta_v \;\; K_Q \right]^T$$

where v_o is the total baseline blood volume and from which the arterial and venous volumes are nominally set at $v_{a0} = 0.3v_0$ and $v_{v0} = 0.7v_0$. Table 2-1 lists the inferred generator model parameters.

Table 2-1: CC model parameters for subject generator model. A diagonal covariance was used during inferencing. The mean and variance values are of the form $(u, \sum)$.

v_o	H_0	Q_0	P_{a0}	K_v	K_a/K_v	K_F
(2.8, 1.3) L	(0.26, 0.12) %	(4.4, 3.39) L/min	(83, 74) mmHg	(5.9, 1.2)	(13, 12)	(0.11, 0.038)
α_I	α_H	K_h	τ_R	K_R	β_v	K_Q
(1.3, 0.13)	(1.3, 0.22)	(61, 12)	(33, 7.4)	(0.64, 0.043)	(1.7, 0.067)	(0.015, 0.002)

2.3 Hemorrhage Model

The CC model and generative model provide a range of physiological characteristics in response to specified hemorrhage. To fully simulate a trauma scenario, further modeling is required to capture how trauma patients lose blood, reach hemostasis, and rebleed.

In this work, uncontrolled blood loss is modeled as a pressure driven outflow from a container. The blood loss is a function of MAP, such that as MAP decreases so does the loss rate. This is done by creating a blood loss "resistance" from an initial hemorrhage rate, which converts a subject's MAP to fluid loss rate:

$$J_H^* = \frac{P_a}{R_H} \tag{2.13}$$

$$R_H = \frac{P_{a0}}{J_{H0}} \tag{2.14}$$

where R_H is the hemorrhage "resistance", J_H^* is the uncontrolled hemorrhage rate, and J_{H0} is the initial hemorrhage rate [12].

Uncontrolled hemorrhage does not continue indefinitely, as the body's natural hemostasis functions will form clots. To realize this behavior, a simple but realistic time dependent clotting model was used. After MAP drops into a hypotensive region clotting starts, at which point the hemorrhage rate decreases over ten minutes until it is zero. This setup can be shown as:

$$C = \begin{cases} 1, & P_a > P_{hyp} \\ 1 - \dfrac{t}{t_{clot}}, & P_a < P_{hyp}\ and\ t < t_{clot} \\ 0, & P_a < P_{hyp}\ and\ t > t_{clot} \end{cases} \tag{2.15}$$

$$J_H = CJ_H^* \tag{2.16}$$

where P_{hyp} is the hypotensive threshold, t is the time since the hypotensive threshold was crossed, t_{clot} is the time for the clot to fully form, and C is a clotting factor that represents how much of the clot has formed [12].

Formed clots have a chance to break, causing rebleeding. Indeed, avoiding rebleeding is an important goal in DCR. Rebleeding occurs when the blood pressure increases to a point where clots rupture. For this work, when rebleeding occurs the clotting factor C is reset to 1 and the uncontrolled hemorrhage is set back to its initial value [12].

When evaluating the algorithms proposed in this work, the simulation is defined such that there is always an initial external hemorrhage and a possible internal hemorrhage. This scenario has been previously used when evaluating DCR procedures [18]. After a period, the external hemorrhage is arrested and fluid resuscitation begins. The internal hemorrhage (if there is one) remains uncontrolled. This scenario is illustrated in figure 2-1, showcasing the hemorrhage modeling.

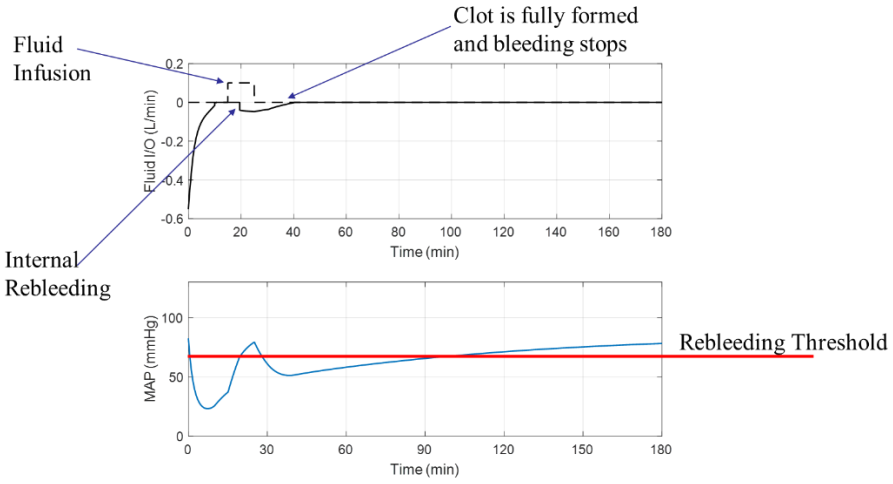

Figure 2-1: Example MAP response, showing hemorrhage and rebleed behavior.

A range of values for initial external hemorrhage, initial internal hemorrhage, and decompensation factors were used when simulating virtual patient responses. These are summarized in table 2-2.

Table 2-2: Hemorrhage and decompensation distributions [18]*.

External J_{H0}	$\sim U(.1,.9)^*$ L/min
Internal J_{H0}	$\sim U(0,.1)^*$ L/min
Decompensation D	$\sim U(0,1)$

2.4 Simple Lumped Model

The SL model is a lumped-parameter model of blood volume kinetics. It is ultimately a simplification of the CC model. The arterial and venous blood volumes are grouped into a single state:

$$\dot{\Delta v} = J_I - J_H - J_F \tag{2.17}$$

where $\Delta v = v - v_0$ is the change in total blood volume from an initial value v_0. The rate of fluid exchange from the tissue space J_F is the same as in equation 2.6, which leads to the total blood volume dynamics:

$$\ddot{\Delta v} + K\dot{\Delta v} = \dot{J_I} - \dot{J_H} + K\left(\frac{1}{1+\alpha_I}J_I - \frac{1}{1+\alpha_H}J_H\right) \tag{2.18}$$

The hurdle associated with using total blood volume as a state is relating it to MAP. From cardiovascular physiology, MAP is the product of cardiac stroke volume and arterial elastance:

$$P_a = V_S E_A \tag{2.19}$$

where V_S is the cardiac stroke volume and E_A is the arterial elastance. The change in MAP can be shown as:

$$\Delta P_a = V_S E_A - V_{s0} E_{A0} = \Delta V_s E_A + \Delta E_A V_{s0} \tag{2.20}$$

where $\Delta P_a = P_a - P_0$, $\Delta V_s = V_s - V_{s0}$, $\Delta E_A = E_A - E_{A0}$; and P_0, V_{s0}, E_{A0} are the initial values of MAP, cardiac stroke volume, and arterial elastance respectively. We can assume that the change in stroke volume and change in total blood volume are proportional, while the change in arterial elastance and change in total volume are inversely proportional:

$$\Delta V_s = K_{V_s} \Delta v \tag{2.21}$$

$$\Delta E_A = -K_{E_A} \Delta v \tag{2.22}$$

where K_{V_s} and K_{E_A} are the gains representing the proportional relationships. The change in pressure can then be shown as a function of total blood volume:

$$\Delta P_a = \left(K_{V_S} E_A - K_{E_A} V_{S_0}\right) \Delta v \tag{2.23}$$

The connection between pressure and volume, $\left(K_{V_S} E_A - K_{E_A} V_{S_0}\right)$, can be defined as a single time varying parameter $K_P(t)$, giving the key relationship:

$$\Delta P_a = K_P(t) \Delta v \tag{2.24}$$

The $K_P(t)$ parameter considers changes in cardiac stroke volume, cardiac regulation, and physiological state and yields the change in MAP. With $K_P(t)$ now defined, the complete SL model can be displayed as:

$$\Delta \ddot{P}_a + K \Delta \dot{P}_a = K_P(t) \left[\dot{J}_I - \dot{J}_H + K \left(\frac{1}{1 + \alpha_I} J_I - \frac{1}{1 + \alpha_H} J_H \right) \right] \tag{2.25}$$

It is assumed that $K_P(t)$ is slowly changing, making the terms including the derivatives of $K_P(t)$ negligible. This modeling decision was verified as accurate in Alsalti et. al. [15].

Chapter 3: Blood Pressure Prediction Algorithm

The BPPA takes previously collected blood pressure (MAP specifically) data from a subject and forecasts the subject's blood pressure into the future. It does this online, meaning for each newly collected MAP data point a new and more accurate prediction is made. First in this chapter, the envisioned decision support system of which the algorithm would be a part, is discussed. Then the following sections of the chapter detail the algorithm itself.

3.1 Decision Support for Hemorrhage Resuscitation

The concept behind using a DSS for resuscitation is that automating the fluid administration process without removing the physician from the feedback loop would increase the settings in which the automation could be used (i.e. in the field rather than only in hospitals) and the reduce the demand for robustness, assisting in achieving regulatory certification. The decision support algorithm envisioned is one which takes prior data and new data as it is measured, personalizes a transparent physics-based model to that data, predicts when blood pressure will drop below a certain threshold with the model, and suggests the optimal timing and size of a fluid bolus to correct the pressure. The BPPA does all of this with the exception of the optimal bolus calculation.

A realistic scenario in which the BPPA (or a DSS) would be applicable is after major blood loss has occurred and all external hemorrhages have been controlled [18]. Starting from this point, the BPPA would collect MAP data and begin individualizing itself to the patient. Since external hemorrhage is controlled, it is reasonable to expect some standard bolus would be given to correct blood pressure to near the hypotensive

threshold or just above [15][18]. It is after this bolus has been administered that predicting blood pressure becomes useful. Here physicians would like to know if MAP is decreasing below a target point and when it will reach that point. Therefore, predictions are not made until after the first bolus is supplied. This procedure is illustrated in figure 3-1.

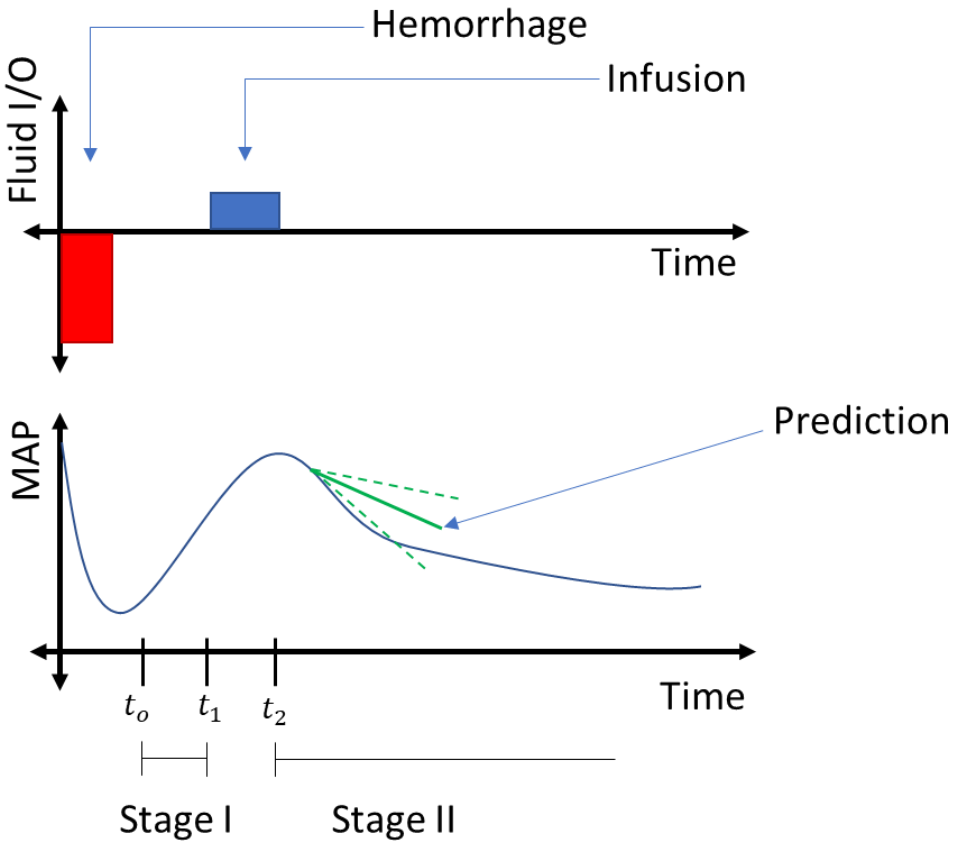

Figure 3-1: Diagram illustrating blood pressure prediction protocol. Data collection starts at t_0. From t_o to t_1 is considered stage I. At t_1 the standard bolus is given. When the bolus finishes at t_2, pressure predictions are made. t_2 until the next bolus is stage II.

A key part of the prediction algorithm is that it adapts to individual subjects. In order for this to occur, the system must be persistently excited to some extent. This is another reason as to why prediction can only begin after an initial bolus. By collecting

MAP data from before, during, and after a bolus, we can identify system parameters and personalize the algorithm to an individual.

While it is understood that all external hemorrhage is controlled, it is possible for uncontrolled internal hemorrhage to be present. Due to coagulopathy or rebleeding, undetected blood loss may be present when the prediction algorithm is implemented. These disturbances are assessed in the evaluation of the BPPA.

3.2 Predicting Blood Pressure

3.2.1 Prediction Model

The goal of the prediction algorithm is to fit the SL model to collected MAP data and then simulate the model into the future. By rewriting equation 2.25, we get a version of the SL model conducive for Kalman filtering:

$$\Delta\dot{P}_a = K_P J_I + K_p K r_F - K\Delta P_a \tag{3.1}$$

3.2.2 The Extended Kalman Filter

Extended Kalman filtering is used to recursively estimate the parameters of equation 3.1 and is explained briefly here. The Kalman filter (KF) is the closed form solution of the marginal posterior distribution of a state (x) at each time step (k) given all previous measurements (y):

$$p(x_k|y_{1:k}) \tag{3.2}$$

The KF can also be interpreted as an observer whose gain (or poles) have been chosen optimally, given information about the noise processes present. It is important to note that the Kalman filter is only an exact solution for linear gaussian dynamic and measurement models. An extended Kalman filter (EKF) expands the KF to nonlinear

systems by making the assumption that the true state is sufficiently close to the estimated state. This lets the error dynamics be accurately represented by a first order expansion of the nonlinear system about a perturbed state [19].

Consider a discrete nonlinear model of the form:

$$x_k = f(x_{k-1}) + w_{k-1} \tag{3.3}$$

$$y_k = h(x_k) + v_k \tag{3.4}$$

where x is a state vector, y is an output vector of interest, $f(x)$ & $h(x)$ are continuously differentiable and possibly nonlinear functions, $w_{k-1} \sim N(0, Q_{k-1})$ is the process noise, and $v_k \sim N(0, R_k)$ is the measurement noise. We use extended Kalman filtering to estimate $p(x_k|y_{1:k})$, by first initializing our estimate of the state vector mean and covariance:

$$\hat{x}_{k-1}^{-} = \hat{x}_0 \tag{3.5}$$

$$P_{k-1}^{-} = P_0 \tag{3.6}$$

where $\hat{x}$ is the estimated state vector, P is the covariance matrix, and k is the current time step. After the initialization step, the rest of the filter proceeds recursively. First the states are propagated forward in time:

$$\hat{x}_k^{-} = F_{k-1}\hat{x}_{k-1} \tag{3.7}$$

$$P_k^{-} = F_{k-1}P_{k-1}F_{k-1} + Q \tag{3.8}$$

where Q is the covariance of the process noise w. F_{k-1} is the partial derivative of the transition matrix evaluated at the prior iteration:

$$F_{k-1} = \frac{\partial \mathrm{f}}{\partial x}\Big|_{\hat{x}_{k-1}} \tag{3.9}$$

Then the Kalman gain is derived:

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R)^{-1} \tag{3.10}$$

where K_k is the Kalman gain, and R is the covariance matrix of the measurement noise v_k. H_k is the partial derivative of the output function at the current iteration:

$$H_k = \frac{\partial h}{\partial x}\Big|_{\hat{x}_k} \tag{3.11}$$

And finally, the states are updated with the measurement error:

$$\hat{x}_k = \hat{x}_k^- + K_k(y_k - H_k \hat{x}_k^-) \tag{3.12}$$

$$P_k = (I - K_k H_k) P_k^- \tag{3.13}$$

The extended Kalman filter solution to the posterior of the state is:

$$p(x_k | y_{1:k}) = N(\hat{x}_k, P_k) \tag{3.14}$$

3.2.3 Multi-Stage Estimation

As mentioned previously, data collection for the BPPA can begin soon after any external hemorrhage is stopped, and predictions can be made after the initial bolus is administered. Over this period (t_0 and onward in figure 3-1), not all parameters and states of the SL model can be estimated at every time instance. Therefore, instead of running an EKF over the entire measurement period, two EKFs are employed at separate set stages.

The first stage is from when the external hemorrhage has stopped until the initial bolus is given (t_0 to t_1 in figure 3-1). Here we do not expect to make predictions and are only looking to estimate the K parameter. When no infusion is present, the SL model reduces to:

$$\Delta \dot{P}_a = K_p K r_F - K \Delta P_a \tag{3.15}$$

An augmented state vector is created such that $x = [x_1,\ x_2,\ x_3]^T$, where $x_1 = \Delta P_a$, $x_2 = K$, and $x_3 = K_P r_F$. The parameters' dynamics are set as random walk, and the resulting nonlinear dynamics for the EKF become:

$$\dot{x_1} = -x_1 x_2 + x_2 x_3 + q_1 \tag{3.16}$$

$$\dot{x_2} = q_2 \tag{3.17}$$

$$\dot{x_3} = q_3 \tag{3.18}$$

where q_1, q_2, and q_2 are independent continuous gaussian noise processes. Over this period, $K_P r_F$ behaves as a time-variant parameter, since the r_F dynamics are zero when no infusion or hemorrhage is present. r_F will, though, change significantly during the initial infusion (t_1 to t_2 in figure 3-1), meaning the estimate of $K_P r_F$ from first stage is not useful for prediction and is discarded.

During the bolus, no parameters are identifiable (due to the rectangular bolus profile not being persistently exciting). The next point at which information can be learned is as soon as the fluid infusion ends. From here to whenever the next bolus is administered is the second stage (t_2 and onwards in figure 3-1). Over the second stage, predictions are made, meaning the state (ΔP_a) estimate and the two parameter (K, $K_P r_F$) estimates are all of importance. The SL model simplifies again to equation 3.15, letting us use the same dynamics from equations 3.16, 3.17, and 3.18 in the formulation of the EKF.

The continuous dynamics in equations 3.16, 3.17, and 3.18 can be approximated as discrete using the first-order Euler method. This gives us the discrete formulation compatible with the EKF described in section 3.2.2:

$$x_{1,k} = x_{1,k-1} + \delta t\left(-x_{1,k-1} x_{2,k-1} + x_{2,k-1} x_{3,k-1}\right) + w_{1,k-1} \tag{3.19}$$

$$x_{2,k} = x_{2,k-1} + w_{2,k-1} \tag{3.20}$$

$$x_{3,k} = x_{3,k-1} + w_{3,k-1} \tag{3.21}$$

where δt is a sufficiently small time step.

The two estimation stages are linked through the final estimate of K from the first stage being used to initialize the estimate of K for the second stage. Essentially, the first EKF serves to initially estimate the K parameter so that the second EKF can begin its estimation closer to the true parameter value. This will help the second EKF converge quicker and allow for more accurate predictions to be generated.

3.2.4 Forecasting Blood Pressure

Blood pressure prediction begins as soon as the initial bolus ends. At this point the second stage EKF is applied to all incoming blood pressure measurements. The EKF provides a new estimate of the MAP and model parameters, $K_P r_F$ and K, for each MAP measurement. Future MAP is predicted by taking those estimates, sampling from their distributions, simulating each sample until some future time, and then finding the mean and standard deviation of the sample MAPs at that time. Thus, a nominal prediction with a quantification of uncertainty is derived. Tables 3-1 and 3-2 summarize the BPPA process over both stages.

Table 3-1: Summary of BPPA algorithm as implemented over stage I (see figure 3-1).

Stage I	
Input: $K_0, K_p r_{F_0}, \Delta \mathrm{P}_{a0}, \sigma^2_{K_0}, \sigma^2_{K_p r_{F_0}}, \sigma^2_{\Delta \mathrm{P}_{a0}}, Q, R, y_{1:N}$	{Initial guesses for states and parameters, noise matrices, stage I measurements}
Output: K_{stageI}, σ^2_K	{K estimate and variance}
$x_0 \leftarrow \left[K_0, K_p r_{F_0}, \Delta \mathrm{P}_{a0} \right]^{\mathrm{T}}$ $P_0 \leftarrow diag(\sigma^2_{K_0}, \sigma^2_{K_p r_{F_0}}, \sigma^2_{\Delta \mathrm{P}_{a0}})$	{Initialize EKF}
loop from 1 to *N*	{Over all measurements from stage I}
$x_k^-, P_k^- \leftarrow Propagate(x_{k-1}, P_{k-1}, Q)$	{Eq. 3.7, 3.8}
$x_k, P_k \leftarrow Update(x_k^-, P_k^-, R, y_k)$	{Eq. 3.10, 3.11, 3.12}
end loop	
return $[x_N]_1, [P_N]_{11}$	

Table 3-2: Summary of BPPA algorithm as implemented over stage II (see figure 3-1).

Stage II	
Input: $K_{stageI}, K_p r_{F_0}, \Delta \mathrm{P}_{a0}, \sigma^2_K, \sigma^2_{K_p r_{F_0}}, \sigma^2_{\Delta \mathrm{P}_{a0}}, Q, R, y_{1:N_P}, t_{future}$	{Initial guesses for states and parameters, initial noise matrices, measurements up until prediction is made, future time to predict to}
Output: $\Delta P_{a,future}, \sigma_{future}$	{MAP Prediction}
$x_0 \leftarrow \left[K_{stageI}, K_p r_{F_0}, \Delta \mathrm{P}_{a0} \right]^{\mathrm{T}}$ $P_0 \leftarrow diag(\sigma^2_K, \sigma^2_{K_p r_{F_0}}, \sigma^2_{\Delta \mathrm{P}_{a0}})$	{Initialize EKF}
loop from 1 to N_P	{Over all measurements from stage II}
$x_k^-, P_k^- \leftarrow Propagate(x_{k-1}, P_{k-1}, Q)$	{Eq. 3.7, 3.8}
$x_k, P_k \leftarrow Update(x_k^-, P_k^-, R, y_k)$	{Eq. 3.10, 3.11, 3.12}
end loop	
$X = \{X_1, X_2, \ldots, X_{100}\} \sim N(x_{N_P}, P_{N_P})$	{Sample from EKF solution}
$P_{future} \leftarrow Simulate(X, t_{future})$	{Simulate Eq 3.1 until future time}
$\Delta P_{a,future} \leftarrow mean(P_{future})$	{Find mean estimate}
$\sigma_{future} \leftarrow std(P_{future})$	{Find standard deviation of estimate}
return $\Delta P_{a,future}, \sigma_{future}$	

Chapter 4: Evaluating the Blood Pressure Prediction Algorithm

In this chapter the in-silico evaluation of the BPPA is presented. The first section explains the simulation scenario implemented on the generated virtual subjects. The following two sections show the performance of the prediction algorithm without any disturbances and the performance when rebleeding is present.

4.1 Simulation Scenario

100 virtual patients were generated as specified in section 2.2. For each patient an initial external hemorrhage, an initial internal hemorrhage, and decompensation factor were sampled from the distributions in table 2-2. Each patient was simulated starting from their healthy baseline MAP with a hemorrhage dictated by the model shown in section 2.3. After 10 minutes the external hemorrhage was stopped (the internal hemorrhage remained uncontrolled). After 15 minutes an initial bolus of 0.1 L/min over 5 min was given. The simulation continued for 35 minutes after the bolus was given. The BPPA was used to predict blood pressure over this 35-minute period.

It is assumed the MAP is measured with invasive monitoring. This is typically done by inserting a small-bore catheter connected to an electronic pressure transducer into a peripheral artery. These systems in general have an accuracy of 5 mmHg or better [20]. Therefore, for the BPPA evaluation the MAP signal was corrupted with 5 mmHg gaussian noise. The measurement noise R matrices for both EKFs were adjusted accordingly.

4.2 No Disturbance Performance

We first assess the BPPA without disturbances. It is assumed that the uncontrolled internal bleeding clots naturally, and the clots do not break when the blood pressure rises from the initial bolus.

4.2.1 Single Virtual Patient

To begin, we examine a single representative virtual patient to visualize the performance of the BPPA. Figure 4-1 shows the EKF estimates over the second stage (t_2 and onwards in figure 3-1). From the MAP state estimate, it is evident that the EKF successfully filters the measurement noise while capturing any error with a two standard deviation (2σ) bound. For reference, the parameter estimates are compared to their corresponding values in the CC model, although they are not identical and some error should be expected. They do show, though, how the filter becomes more confident over time, suggesting that the BPPA's accuracy is time dependent.

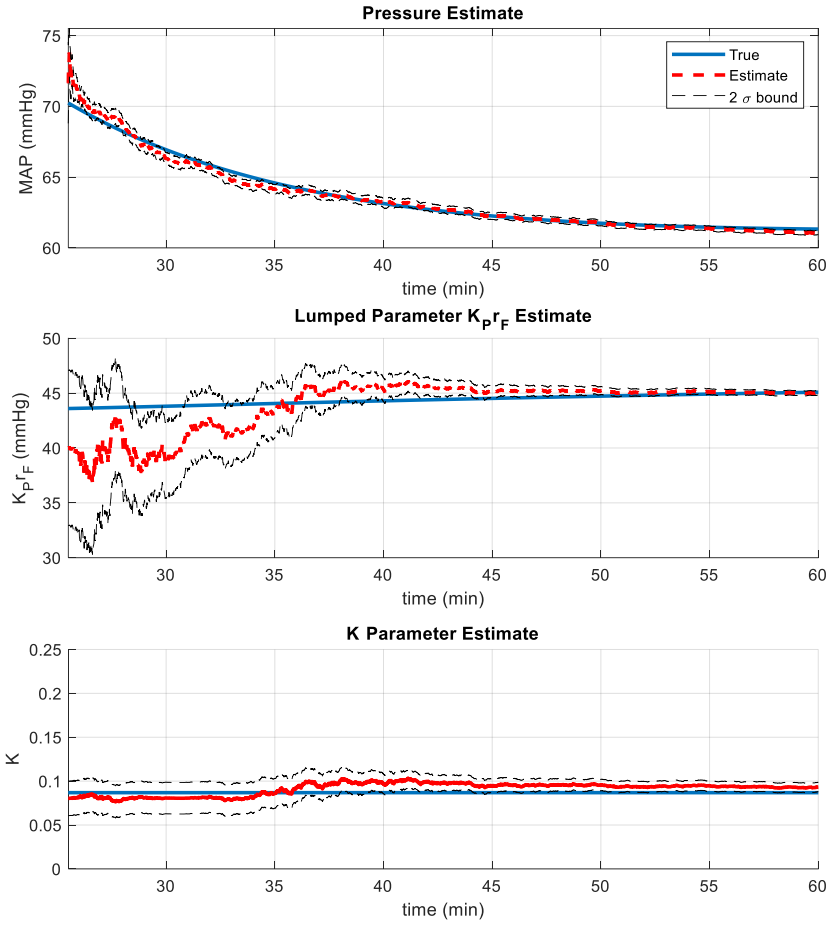

Figure 4-1: EKF estimates for a representative virtual patient over stage II.

The BPPA's performance is visualized in Figure 4-2, which shows selected 5-, 10-, and 15-minute predictions for a representative virtual patient. It is clear the BPPA does successfully forecast blood pressure, although not completely without error. That error, though, is captured well by the 2 standard deviation uncertainty bound also shown. As was hinted in figure 4-1, the estimates begin quite uncertain and as the EKF converges become more accurate.

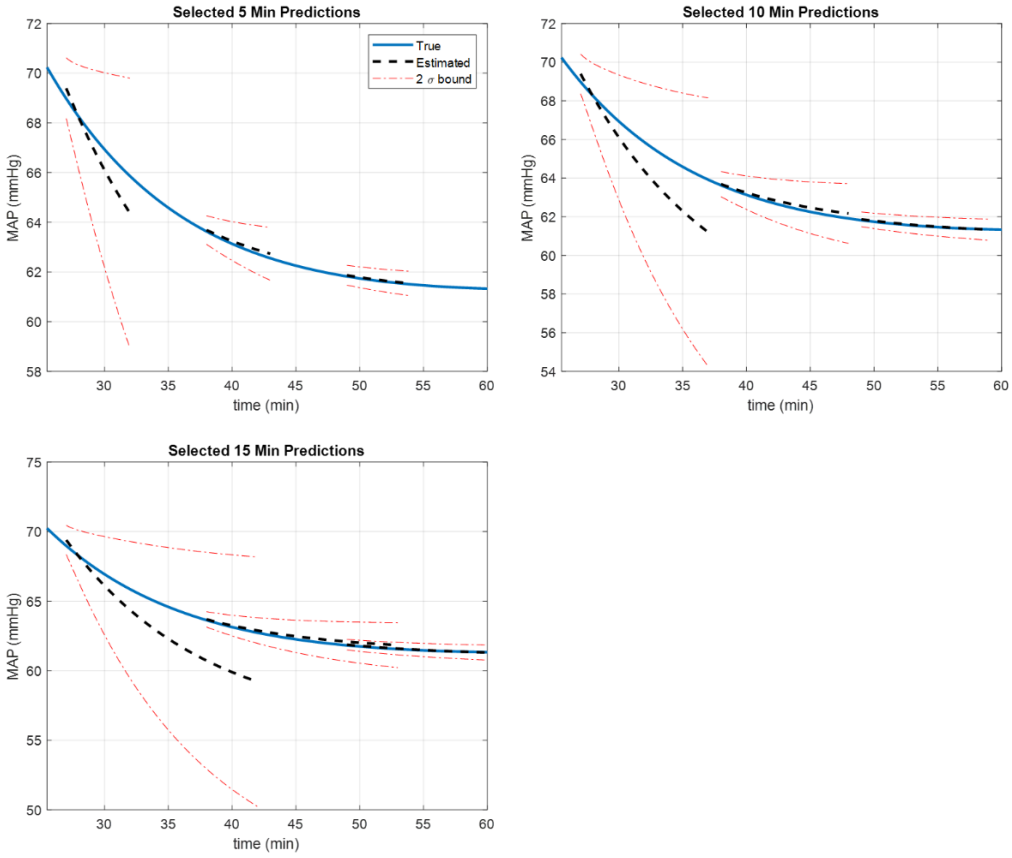

Figure 4-2: MAP predictions for a representative virtual patient.

Figure 4-3 further illustrates the time dependence of the BPPA's performance. It shows the errors of the predictions made at every time instance over the prediction period. The EKF clearly requires time to converge, during which the parameter estimates are uncertain, prompting predictions that may lack accuracy. Once the EKF does converge the prediction error lowers, and eventually converges to near zero as the true blood pressure begins to plateau.

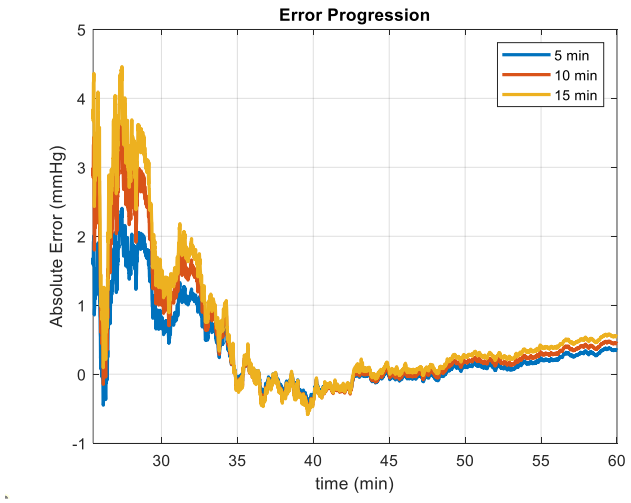

Figure 4-3: Error progression over prediction period for all 3 prediction lengths for a single virtual patient.

4.2.2 Virtual Patient Cohort

While the single virtual patient analysis is effective at visualizing the BPPA, analyzing the BPPA over a cohort of virtual patients is required to evaluate its effectiveness. The normalized prediction error distribution over 100 virtual patients is shown in figure 3-6 via a box and whisker plot. Root mean squared error (RMSE) is used to average the error across all time instances in the prediction period (recall that the BPPA makes a new prediction for each new measurement). The normalization component comes from dividing the prediction error by total change in MAP over the 5, 10, or 15 minutes that the prediction was made, before the RMSE is derived. Therefore, a normalized error of 1 is roughly equivalent to the prediction error being equal to the total shift in MAP. This prevents predictions over stagnant blood pressure spots from being over-weighted. This same figure has the performance of a "dummy" algorithm, which predicts that the MAP will not change from the last MAP estimate made by the EKF.

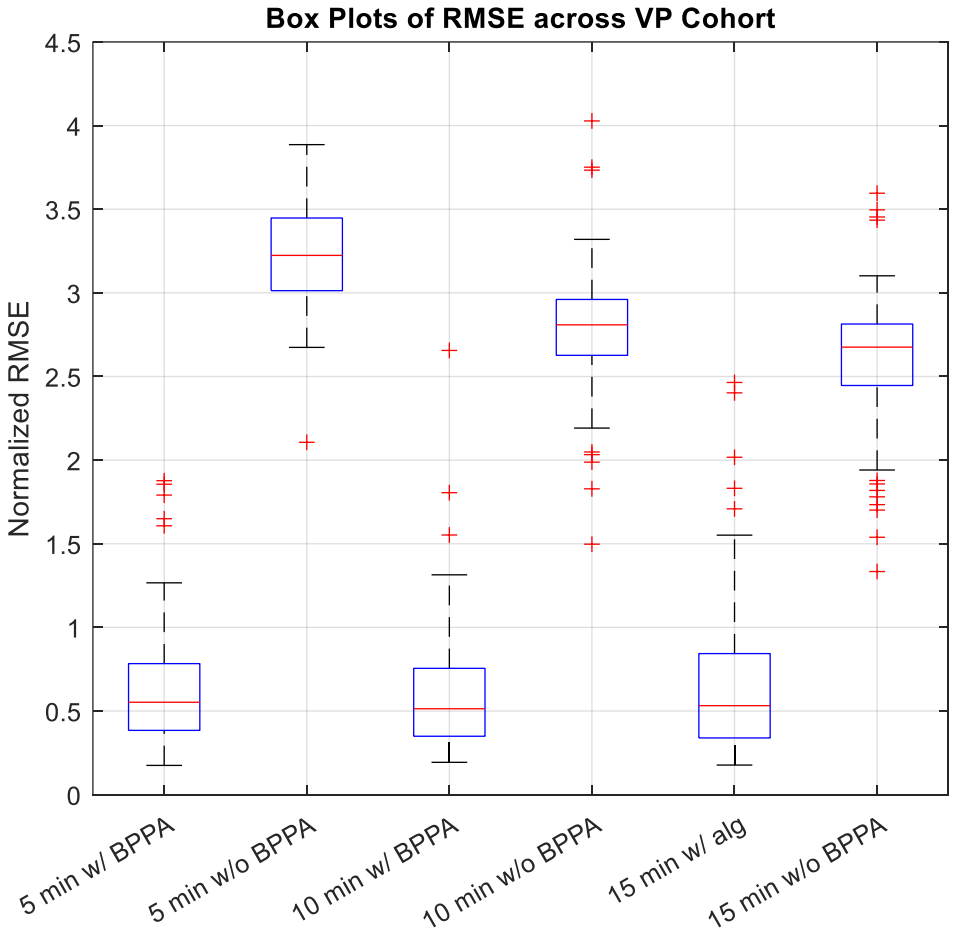

Figure 4-4: Distributions of normalized RMSE for BPPA across the prediction period. The distributions of errors for when the BPPA is not used are shown as well.

Figure 4-4 shows several positive results. The first is that the BPPA substantially outperforms the "dummy" algorithm, demonstrating that the BPPA is consistently accurate and useful. For the majority of virtual patients, the BPPA's normalized error is less than 1 for both short and long predictions, suggesting accuracy does not deteriorate relative to prediction lengths. We do see that some outlier virtual patients have a normalized error of greater than 1. This implies that in a select few scenarios the BPPA may lack the accuracy required for DSSs.

4.3 Rebleed Performance

To evaluate the BPPA's performance with disturbances, rebleeding is introduced. When the blood pressure rises past a threshold clots break, and the internal hemorrhage reoccurs (this was true for all 100 VPs).

4.3.1 Single Virtual Patient

Figure 4-5 shows the EKF estimates over the second stage of a single virtual patient, while rebleeding is ongoing. It is immediately clear that the EKF is compensating for the unknown disturbance by shifting its parameter estimates. The uncertainty bounds do not reflect this compensation, and the EKF becomes confident in the shifted estimates, despite their distance from the analogous CC model parameters.

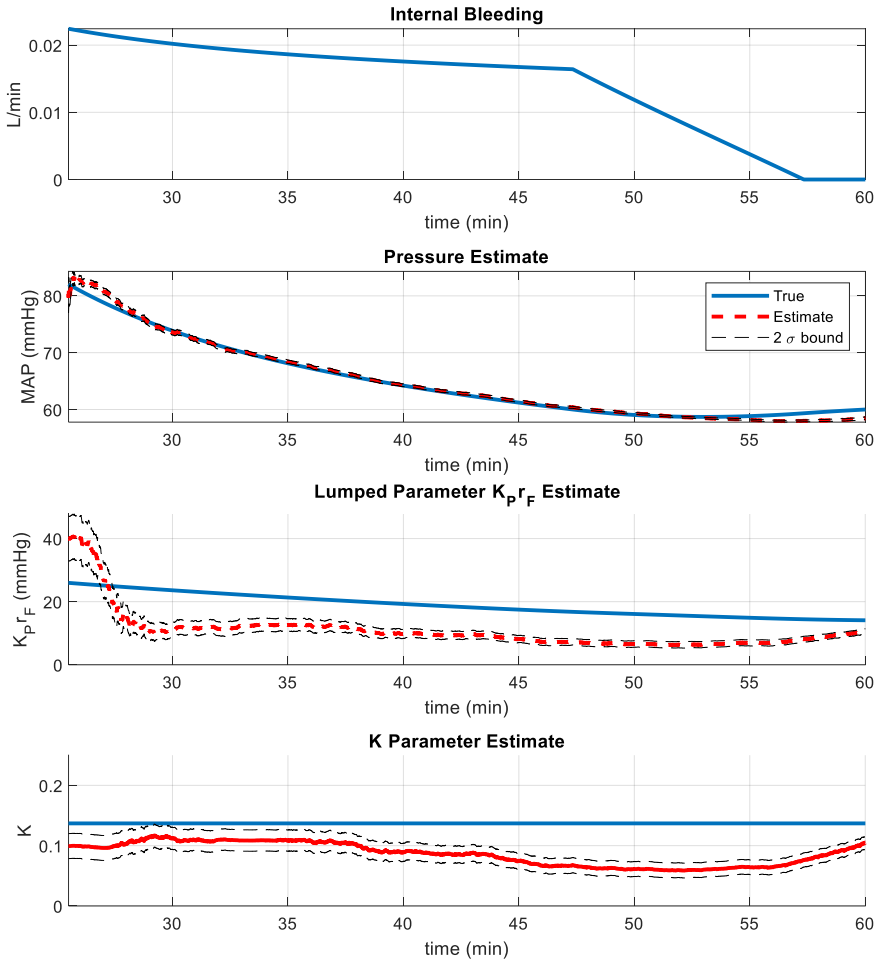

Figure 4-5: EKF estimates, while internal bleeding is occurring, for a representative virtual patient over stage II.

The example BPPA predictions during rebleeding are shown in Figure 4-6. The accuracy has deteriorated and the 2σ bounds no longer are certain to capture the error. Figure 4-7 shows the error progressions over time in more detail. We can see that the EKF again takes time to converge, after which the predictions are relatively accurate. When the internal bleeding begins to lessen (as the clots reform), the predictions become poor again.

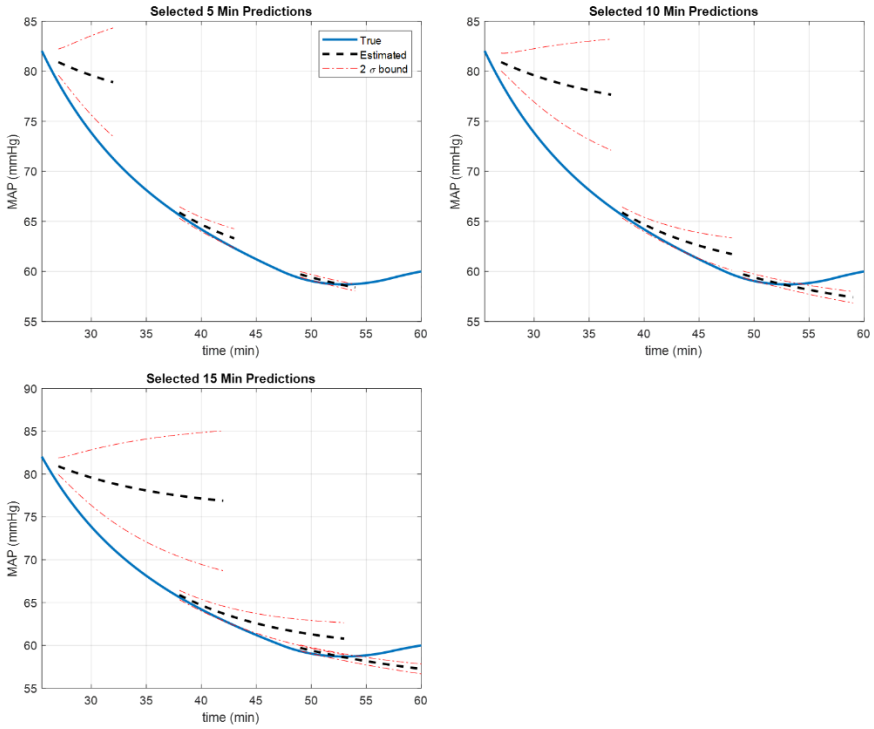

Figure 4-6: MAP predictions for a representative virtual patient, during rebleeding.

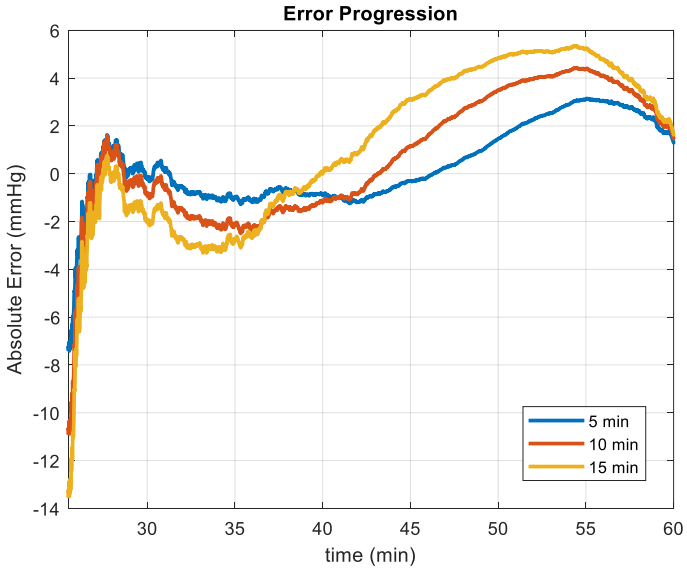

Figure 4-7: Error progression over prediction period for all 3 prediction lengths, during rebleeding.

4.3.2 Virtual Patient Cohort

Figure 4-8 shows the normalized RMSE values averaged across the prediction period for 100 virtual patients all of which had rebleeding triggered. The BPPA

accuracy has deteriorated substantially compared to the no disturbance results. The 5-minute predictions with the BPPA remain more accurate than the predictions without the algorithm, but the 10- and 15-minute predictions have medians worse than if no algorithm was used. It is also evident that the variance in accuracy has vastly grown.

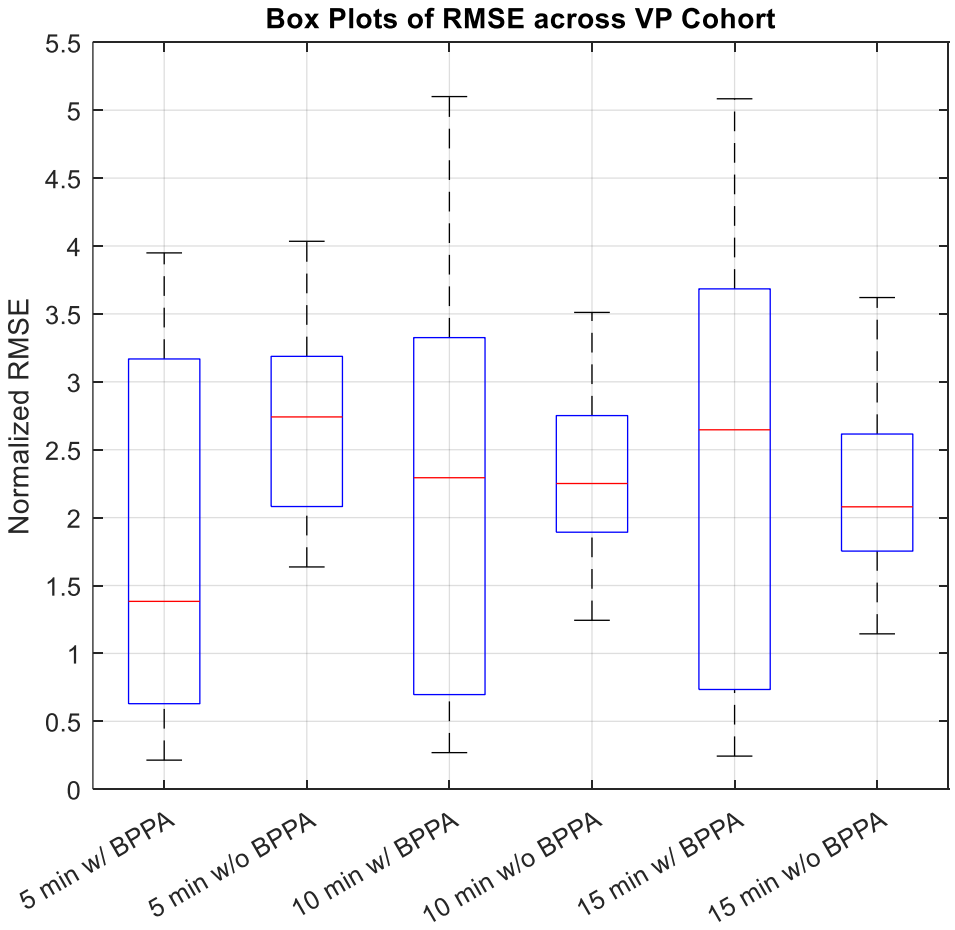

Figure 4-8: Distributions of normalized RMSE for BPPA across the prediction period, when rebleeding is occurring. The distributions of errors for when the BPPA is not used are shown as well.

4.4 Discussion

Overall, the BPPA performed well. It dealt with substantial noise to make accurate predictions and generated uncertainty bounds which captured any error. An important feature of the algorithm is that it was least accurate just after the bolus was

given, as the EKF required time to converge. This is possibly counter-intuitive as this is when the prediction is likely most needed. Unfortunately, the EKF must take some time to converge, and cannot start while the bolus is being given. Estimation over the input period becomes much more complex. To identify all relevant parameters, the input profile must be well known and persistently exciting, which is unrealistic. Therefore, we must accept some loss of accuracy initially.

The rebleeding performance was poor. The BPPA was not robust enough to deal with major disturbances. What was promising was that the predictions were relatively accurate after the convergence period and before the bleeding stopped. This suggests that if a hemorrhage detection capability was added to the BPPA then it could adjust itself when bleeding began or ended. Multiple Model Adaptive Estimation (MMAE) that reacts to blood loss is a possible route that could help robustify the BPPA.

Chapter 5: Absolutely Stable Closed-Loop Control

The closed-loop controller presented in this chapter regulates blood pressure to a target level through infusion. A major challenge in designing controls for critical care is the high variance among patients' physiologies. To overcome this hurdle, the controller was designed using an LMI technique such that it ensured the system was absolutely stable. Absolute stability and the requirements for it to be met by a system are discussed first in this chapter, followed by an overview of the observer-based integral controller, and then how LMIs were used for design.

5.1 Closed-Loop Control for Hemorrhage Resuscitation

While the envisioned decision support system described in section 3.1 keeps physicians in the loop, closed-loop controllers change their role to high-level supervision. Thus, these controllers cannot rely on constant physician monitoring, and must be robust enough to perform not only under inter-subject variability, but under common disturbances such as active uncontrolled hemorrhages, rebleeding, and coagulopathy [13].

Work has already been done to develop fluid administration controllers. Most existing controllers, though, are formed from empirical gain tuning and show room for improving efficacy and robustness [13][14]. On the other side of the spectrum, complex adaptive controllers built around control-theoretic models of hemodynamics have been developed. While these have been shown to perform well, they require particular pre-control procedures and hematocrit measurements to reach full efficacy [15]. Absolutely stable control finds a middle ground. A system is absolutely stable when it is

asymptotically stable over a class of nonlinearities that satisfy a sector condition [21]. It is possible to design a controller using computationally efficient LMI techniques to make a system absolutely stable. This would offer robustness guarantees in the context of control theory (important for regulatory certification and clinical use), while using a relatively simple controller.

5.2 Absolute Stability

A system can be assessed for absolute stability when it has a feedback loop connecting a linear plant with a nonlinear element. Figure 5-1 illustrates an absolute stability compatible system.

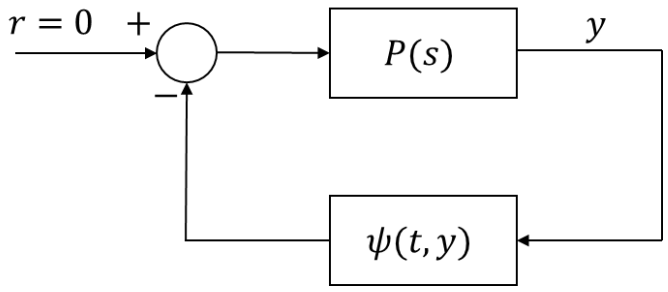

Figure 5-1: System compatible with absolute stability analysis. P(s) is a linear plant. $\psi(t, y)$ is a nonlinear function.

The system can also be represented by:

$$\dot{x} = Ax + Bu \tag{5.1}$$

$$y = Cx \tag{5.2}$$

$$u = -\psi(t, y) \tag{5.3}$$

where $x \in \mathbb{R}^n, y \in \mathbb{R}$, (A, B) is controllable, (A, C) is observable, and ψ is a memoryless, possibly time-varying nonlinearity which is piecewise continuous in t and locally Lipschitz in y.

To evaluate the stability of the system, we must first put a limit on the nonlinearity, called a sector condition. The nonlinearity ψ belongs to sector $[\alpha, \beta]$ if:

$$\alpha y^2 \leq y\,\psi(t,y) \leq \beta y^2, \qquad \forall\, t \geq 0, \qquad \forall\, y \in [a,b] \tag{5.4}$$

where a and b are constants such that $a < 0 < b$. A system is absolutely stable about its origin when it is asymptotically stable for all nonlinearities within a sector.

There are a variety of ways to show that a system is stable across a sector. One method that is particularly useful for design is employing simultaneous Lyapunov functions. Consider the candidate Lyapunov function:

$$V(x) = x^T P x \tag{5.5}$$

where P is positive definite and symmetric. The time derivative of $V(x)$ along trajectories of the system is:

$$\dot{V}(t,x) = x^T(PA + A^T P)x - 2x^T PB\psi(t,y) \tag{5.6}$$

The goal becomes to find a matrix P such that $\dot{V}(t,x)$ is negative definite for all nonlinearities which satisfy the sector condition. We start by defining a time varying gain $k(t)$ to represent the nonlinearity:

$$\psi\big(t, y(t)\big) = k(t)y(t) \tag{5.7}$$

If $\psi\big(t, y(t)\big)$ is within sector $[\alpha, \beta]$, then we know $k(t)$ must be bounded by:

$$\alpha \leq k(t) \leq \beta \tag{5.9}$$

irrespectively of any particular $x(t)$ trajectory. Returning to equation 5.6 the derivative can be shown as:

$$\dot{V}(t,x) = x^T(PA + A^T P)x - 2x^T PBk(t)y \tag{5.10}$$

$$= x^T[P(A - Bk(t)C) + (A - Bk(t)C)^T P]x$$

If the right-hand side of equation 5.10 is negative definite for all time varying gains, then the original Lyapunov function derivative (equation 5.6) is negative definite for all nonlinearities. In other words, to show the system is absolutely stable, we can show:

$$x^T[P(A - Bk(t)C) + (A - Bk(t)C)^T P]x < -\epsilon x^T x, \ \ \forall\, k \in [\alpha, \beta] \quad (5.11)$$

where ϵ is a positive constant. Since the left-hand side of the inequality is linear in k, it attains its maximum value over $[\alpha, \beta]$ at one of the boundaries. Therefore, to show absolute stability we actually only need to show:

$$x^T[P(A - Bk_iC) + (A - Bk_iC)^T P]x < -\epsilon x^T x, \qquad \forall\, i = 1, i = 2 \quad (5.12)$$

where $k_1 = \alpha$ and $k_2 = \beta$.

In summary, the system shown in equations 5.1-5.3 is absolutely stable if the sector condition in 5.9 is satisfied, and there exists a symmetric positive definite matrix P and positive constant ϵ, such that [21]:

$$P(A - Bk_iC) + (A - Bk_iC)^T P < -\epsilon I, \qquad \forall\, i = 1, i = 2 \qquad (5.13)$$

5.3 Observer-Based Integral Control

Observer-based integral control was chosen to regulate blood pressure for two reasons: 1) the state estimates let us choose where the closed loop system poles are placed, 2) the integrator component gave us the ability to track setpoints away from the origin. The pole placement was important for flexibility in adjusting the controller performance. Setpoint tracking was necessary to keep the system nonlinearity from changing signs during the controller's operation period, which was a requirement for absolute stability analysis (section 5.4 explains this further).

The controller was designed around the SL model as specified in section 2.4. The SL model can be interpreted as a linear plant with a nonlinear element. The linear

portion of the SL model can be written in state space form (as in equations 5.1 and 5.2), where $x = [\Delta v,\ r_F]^T$, with

$$A = \begin{bmatrix} -K & K \\ 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 1 \\ \dfrac{1}{1+\alpha_u} \end{bmatrix}, C = [1 \quad 0]$$

We can define the nonlinear element as:

$$z = \psi(y,t) = K_P(t)y \tag{5.14}$$

where $z = \Delta P_a$.

The observer-based feedback can then be built around the SL model plant, starting with a Luenberger observer:

$$\dot{\hat{x}} = A\hat{x} - Bu + F_o(z - \bar{\psi}Cx) \tag{5.15}$$

where $\hat{x}$ is the state estimate, F_o is the observer gain, and $\bar{\psi}$ is a constant approximation of $\psi(y,t)$. The estimated state feedback is combined with an integrator state, x_N, to form the input to the system:

$$u = -F_c\hat{x} + F_e x_N \tag{5.16}$$

where F_c is the estimated state feedback gain and F_e is the integrator gain. The integrator state has the dynamics:

$$\dot{x_N} = r - z = r - K_P(t)y \tag{5.17}$$

where r is a reference signal. Figure 5-2 shows the closed-loop system in block diagram form.

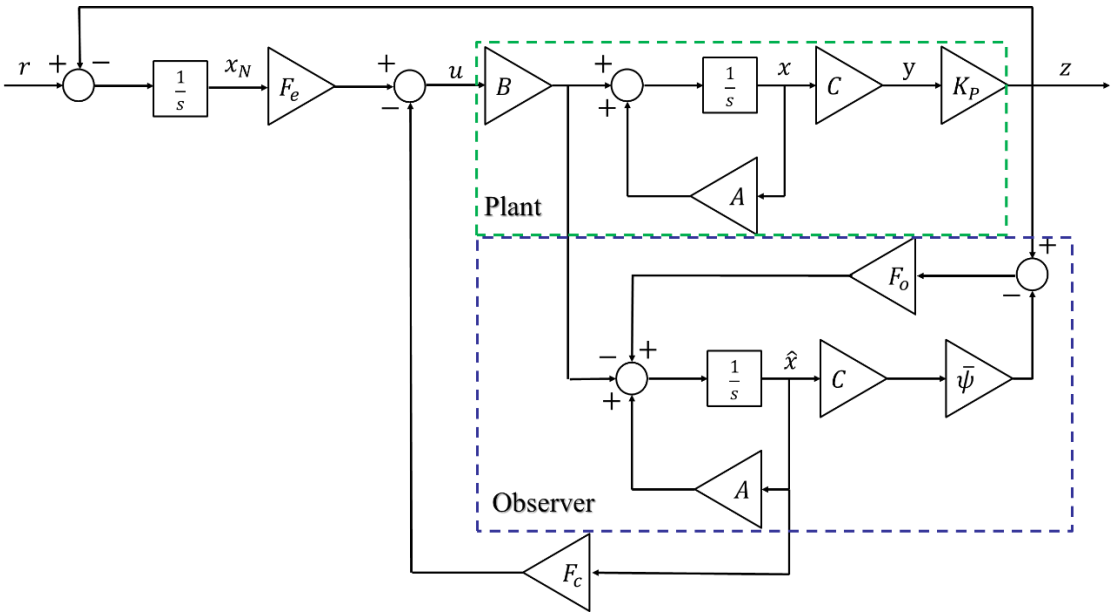

Figure 5-2: Block diagram of closed-loop system including the SL model plant and observer-based integral controller.

If the reference signal is set to zero, the closed-loop dynamics can be aggregated into a linear plant with nonlinear feedback:

$$\begin{bmatrix} \dot{x} \\ \dot{\tilde{x}} \\ \dot{x_N} \end{bmatrix} = \bar{A} \begin{bmatrix} x \\ \tilde{x} \\ x_N \end{bmatrix} + \bar{B}z \tag{5.18}$$

$$y = \bar{C}x \tag{5.19}$$

with:

$$\bar{A} = \begin{bmatrix} A - BF_c & BF_C & BF_e \\ \bar{\psi}F_oC & A - \bar{\psi}F_oC & 0 \\ 0 & 0 & 0 \end{bmatrix}, \bar{B} = \begin{bmatrix} 0 \\ -F_o \\ -1 \end{bmatrix}, \bar{C} = [C \quad 0 \quad 0]$$

where $\tilde{x} = x - \hat{x}$, and the nonlinear feedback element is the same as in equation 5.14. Thus, we have a system compatible with absolute stability analysis.

To finish the design, controller and observer gains must be found such that the system remains stable across a defined sector for the nonlinear function or in this case the $K_P(t)$ parameter.

5.4 Design via Linear Matrix Inequalities

Recall that $K_P(t)$ is a time-varying parameter that considers changes in cardiac stroke volume, cardiac regulation, and physiological state. A large portion of inter-subject physiological uncertainty is contained within $K_P(t)$. We would like to design a controller that is stable over this uncertainty.

To begin designing the absolutely stable controller, let $K_P(t)$ satisfy a sector condition:

$$K_{P,min} \leq K_P(t) \leq K_{P,max} \tag{5.20}$$

where $K_{P,min}$ and $K_{P,max}$ are bounds over which we would like to show stability.

The closed loop system described in equations 5.18 and 5.19 is absolutely stable, if there exists a symmetric positive definite matrix P and positive constant ϵ such that:

$$P(\bar{A} + \bar{B}k_i\,\bar{C}) + (\bar{A} + \bar{B}k_i\,\bar{C})^T P < -\epsilon I, \qquad \forall\, i = 1, i = 2 \tag{5.21}$$

where $k_1 = K_{P,min}$ and $k_2 = K_{P,max}$.

The choice of bounds on $K_P(t)$ is critical to whether or not equation 5.21 can be used to show absolute stability. In fact, for this application it was discovered that if $K_{P,min} < 0$ and $K_{P,max} > 0$, there existed no matrix P such that equation 5.21 was satisfied. This is the challenge that drove the use of integral control. Without integral control, $K_{P,min} < 0$ and $K_{P,max} > 0$ would have been a requirement to capture the uncertain in $K_P(t)$. Due to the restriction that the target blood pressure had to be set as the SL model baseline $(v_0, P_{a,0})$, during regulation $K_P(t)$ would approach zero, and, if overshoot occurred, switch signs. Integral control lets us track a reference input (r) away from the system baseline without error. This was used to move the SL model

baseline below the target blood pressure, while still hitting the target using the reference input. As a result, sector condition bounds changed to $K_{P,min} > 0$ and $K_{P,max} > 0$, making it possible for equation 5.21 to be satisfied.

Equation 5.21 can be expanded to be rewritten as follows:

$$P\begin{bmatrix} A-BF_c & BF_C & BF_e \\ (\bar{\psi}-k_i)F_oC & A-\bar{\psi}F_oC & 0 \\ -k_iC & 0 & 0 \end{bmatrix} + \begin{bmatrix} A-BF_c & BF_C & BF_e \\ (\bar{\psi}-k_i)F_oC & A-\bar{\psi}F_oC & 0 \\ -k_iC & 0 & 0 \end{bmatrix}^T P < -\epsilon I \quad (5.22)$$

P can be broken into blocks to facilitate further expansion:

$$P = \begin{bmatrix} P_1 & Q & X \\ Q^T & P_2 & Y \\ X^T & Y^T & P_3 \end{bmatrix} \quad (5.23)$$

Then the fully expanded inequality becomes:

$$S < -\epsilon I, \ \ \forall\, i = 1, i = 2 \quad (5.24)$$

where

$$[S]_{11} = P_1A + A^TP_1 - P_1BF_C - F_C^TB^TP_1 + (\bar{\psi}-k_i)QF_oC + (\bar{\psi}-k_i)C^TF_o^TQ^T - k_iXC - k_iC^TX^T$$

$$[S]_{12} = P_1BF_c + QA + A^TQ - \bar{\psi}QF_oC - F_c^TB^TQ + (\bar{\psi}-k_i)C^TF_o^TP_2 - k_iC^TY^T$$

$$[S]_{13} = P_1BF_e + A^TX - F_c^TB^TX + (\bar{\psi}-k_i)C^TF_o^TY - k_iC^TP_3$$

$$[S]_{21} = Q^TA - Q^TBF_C + (\bar{\psi}-k_i)P_2F_oC - k_iYC + F_c^TB^TP_1 + A^TQ^T - \bar{\psi}C^TF_o^TQ^T$$

$$[S]_{22} = Q^TBF_c + P_2A - \bar{\psi}P_2F_oC + F_c^TB^TQ + A^TP_2 - \bar{\psi}C^TF_o^TP_2$$

$$[S]_{23} = Q^TBF_e + F_c^TB^TX + A^TY - \bar{\psi}C^TL^TY$$

$$[S]_{31} = F_eB^TP_1 + X^TA - X^TBF_c + (\bar{\psi}-k_i)Y^TF_oC - k_iP_3C$$

$$[S]_{32} = X^TBF_c + Y^TA - \bar{\psi}Y^TF_oC + F_eB^TQ$$

$$[S]_{33} = X^TBF_e + F_eB^TX$$

There are computationally efficient numerical LMI solvers that can be applied to equation 5.24. Unfortunately, equation 5.24 is not linear with respect to the unknown matrices: P, F_c, F_e, and F_o. Therefore, to make equation 5.24 into a LMI we must settle on a few compromises.

The first is that we specify a F_c and F_e before solving the LMI. Both of these gains only appear in the S matrix when they are multiplied by a constant matrix and then by a submatrix of the unknown P matrix (e.g. $P_1 B F_c$). By choosing these gains prior to solving the LMIs, the terms they are a part of become linear. This is not necessary for F_o, because the observer gain is only ever directly multiplied by a submatrix of the P matrix (e.g. $(\bar{\psi} - k_i) P_2 F_o C$). This lets us replace the gain and the P matrix with a single representative variable, such as $Z = P_2 F_o$. The solution for F_o can then be found after solving the LMI as follows:

$$F_o = P_2^{-1} Z \tag{5.25}$$

Thus, the S matrix can be made linear. Note that this revises the problem into essentially observer design, as we are now only finding F_o.

The second compromise is that P must be limited, such that $Q = Y = 0$. This restriction is needed to find a unique solution for the observer gain. Without it, F_o appears in terms with Q, Y, and P_2. This would require forming three different lumped variables (e.g. $Z_1 = P_2 F_o$, $Z_2 = Q F_o$, $Z_3 = Y F_o$), and solving the LMI would result in three different observer gain solutions. Setting Q and Y to zero ensures there is only one solution for F_o. As a consequence, the solver becomes conservative, in that it cannot find some P matrices which may satisfy equation 5.21.

With these edits in place, equation 5.24 simplifies to an LMI with variables: P_1, P_2, P_3, X, and Z; where Z is the proposed variable that contains the observer gain, (i.e. $Z = P_2F_o$). After solving the LMI, F_o is found from equation 5.25.

5.5 Absolutely Stable Observer-Based Integral Controller

The absolutely stable observer-based integral controller was designed around the nominal SL plant (found from the same inferencing method described in section 2.2). The controller gains were selected using pole placement such that the dynamics were critically damped. Two sets of controller gains at different speeds were chosen to illustrate the limits of the controller. The sector condition bounds for $K_P(t)$ were chosen by simulating the CC model and finding the maximum and minimum values of the MAP to total blood volume ratio. $\bar{\psi}$ was made the center of the sector. All of these inputs were provided to equation 5.24 and MATLAB's Robust Control Toolbox was used to solve for F_o.

At this point a complication arose. The LMI solver could not find a solution. Therefore, the sector bounds had to be reduced until the LMI solver could find a suitable observer gain. Recall that by making certain elements of the P matrix zero the solution was made conservative. This meant that the solver not finding a gain for the full sector did not prove there was no such F_o that made that system stable. With this in mind, the observer gain found from the reduced sector was tested with the full sector for absolute stability. It was determined the F_o found from the reduced sector did in fact stabilize the system across the full sector.

The two controller gains and their corresponding observer gains found via the LMI method described above are shown in table 5-1. Controller 2 is more aggressive than controller 1.

Table 5-1: Summary of values used to design controllers. Controller 1 is slower than Controller 2.

		Common Inputs	
$K_{P,min}$	(mmHg/L)	20	
$K_{P,max}$	(mmHg/L)	150	
$\bar{\psi}$	(mmHg/L)	85	
		Controller 1	Controller 2
F_c		$[0.20, -0.042]$	$[0.40, -0.20]$
F_e		$3.3 \cdot 10^{-4}$	$6.5 \cdot 10^{-4}$
F_o		$[0.08, 0.11]^T$	$[0.49, 0.37]^T$

Chapter 6: Evaluating the Closed-Loop Controller

In this chapter, the closed-loop controllers from the previous chapter are evaluated in-silico. The scenario simulated over is explained first, followed by the performance of the controllers. The final section discusses the efficacy of the LMI design technique compared to less involved methods.

6.1 Simulation Scenario

The controller at both speeds was evaluated over a setup similar to the BPPA. 100 virtual patients were generated, and each was given an initial external hemorrhage, an initial internal hemorrhage, and decompensation factor sampled from the distributions in table 2-2. Each patient was simulated starting from their healthy baseline MAP with their hemorrhage dictated by the model shown in section 2.3 After 10 minutes the external hemorrhage was stopped. From here 10 minutes passed before the controller was "turned on." All virtual patients were regulated to the same setpoint of 85 mmHg. Importantly, they were not all regulated from the same starting blood pressure nor the same initial state. Rebleeding was not triggered. The controller was not evaluated on any major disturbances.

It was assumed that the MAP is measured with invasive monitoring, and a volumetric infusion pump inputs whole blood. Measurement noise was included, although not to the extent of the BPPA analysis. The MAP signal was contaminated with a zero mean gaussian noise with a standard deviation of 2 mmHg [15]. A six-point moving average digital filter was incorporated into the control algorithms to minimize the effect of the noise.

6.2 Controller Performance

Figure 6-1 shows the performance of a single representative virtual patient. For controller 1 (the slower controller), the response hit the target MAP accurately with little overshoot and minimal steady state error. Controller 2 has a noticeable overshoot, but still ends with little steady state error.

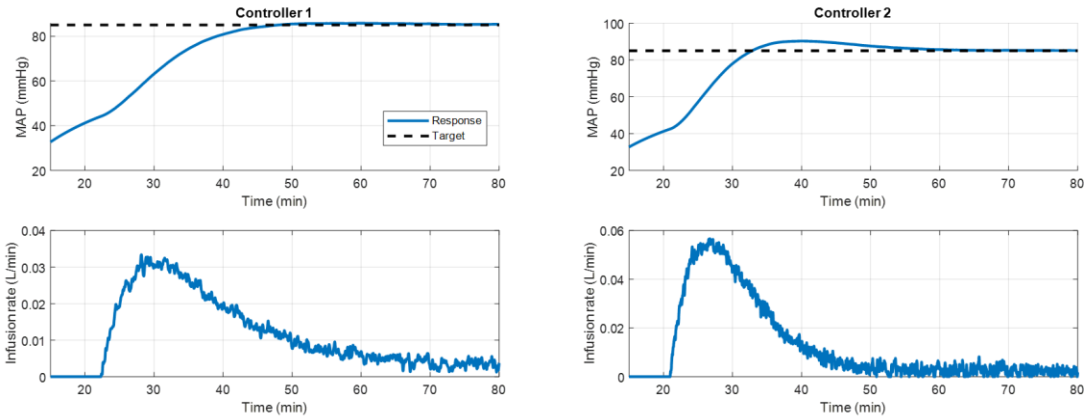

Figure 6-1: MAP responses of a representative virtual patient. Time zero is when the hemorrhage occurs. The controller is activated at 20 minutes.

The responses for all 100 virtual patients are visible in figure 6-2. It is evident that the slower controller performs well across different physiologies, while the faster controller pays a penalty in performance for speed. Table 6-1 summarizes the performance of the controller with several metrics. The Median Performance Error (MDPE) is the median of the performance error, which is as follows:

$$PE = \frac{P_a - P_{a,target}}{P_{a,target}} \times 100\ \% \tag{6.1}$$

where $P_{a,target}$ is the target setpoint (85 mmHg). The Median Absolute Performance Error (MDAPE) is the median of the absolute value of PE. The target overshoot is the maximum MAP relative to the target. Rise time is the time required to reach within 90% of the target.

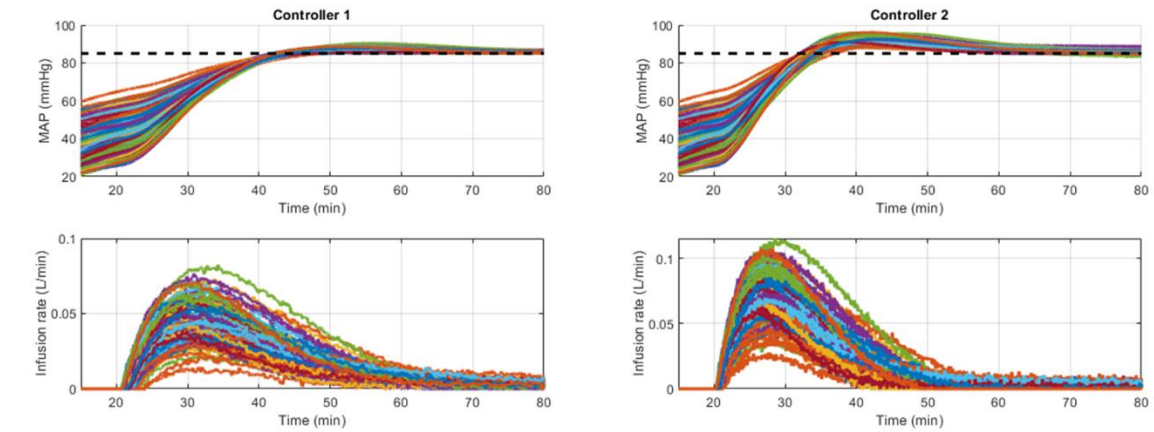

Figure 6-2: MAP responses for all 100 virtual patients

Table 6-1: Performance metric distributions for both controllers. The 2 standard deviation bounds are shown.

	Controller 1	Controller 2
MPDE (%)	0.77 ± 0.86	1.19 ± 1.73
MDAPE (%)	2.27 ± 2.14	4.03 ± 2.58
Target Overshoot (mmHg)	2.15 ± 2.1	7.2 ± 3.55
Rise Time (min)	16.57 ± 1.68	10.44 ± 1.47

The nonlinearity sectors are shown in figure 6-3. As there is no explicit function to plot, the black lines are the blood volume blood pressure pairs from the simulations of the CC model across the control period for all 100 virtual patients. The chosen sector captures the VP nonlinearities well, giving us confidence in the absolute stability of the system.

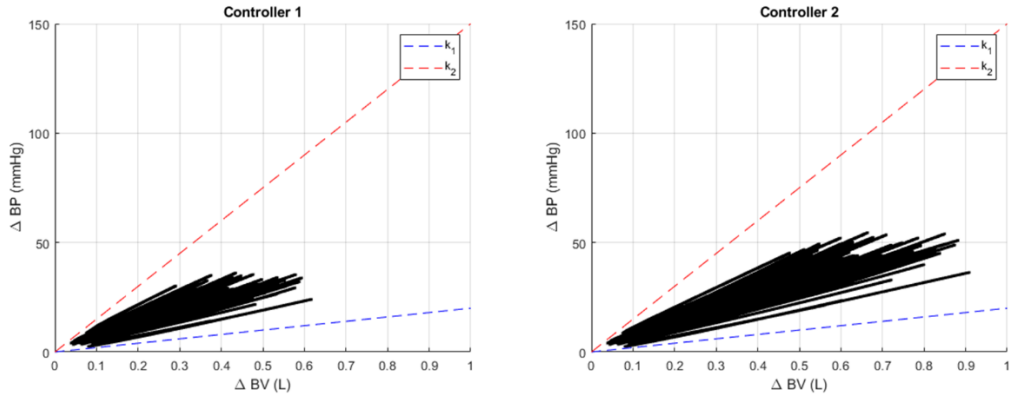

Figure 6-3: VPs' blood volume to blood pressure ratio relative to sector bounds.

6.3 Stability Analysis

A natural question to ask is "Was the LMI design technique really necessary?" There is the possibility that blindly chosen observer gains are absolutely stable, rendering the LMI design method worthless for this application. To analyze this notion, all possible pairings of the elements of the observer gain were assessed for absolute stability across the sector from table 5-1. Instead of testing only the nominal plant model (which had been used for design), 100 SL model virtual patients were generated and evaluated for each observer. Figure 6-4 shows the results. The yellow regions are where 100% of the virtual patients were not able to be determined as absolutely stable. The purple regions are where 100% of the virtual patients were absolutely stable.

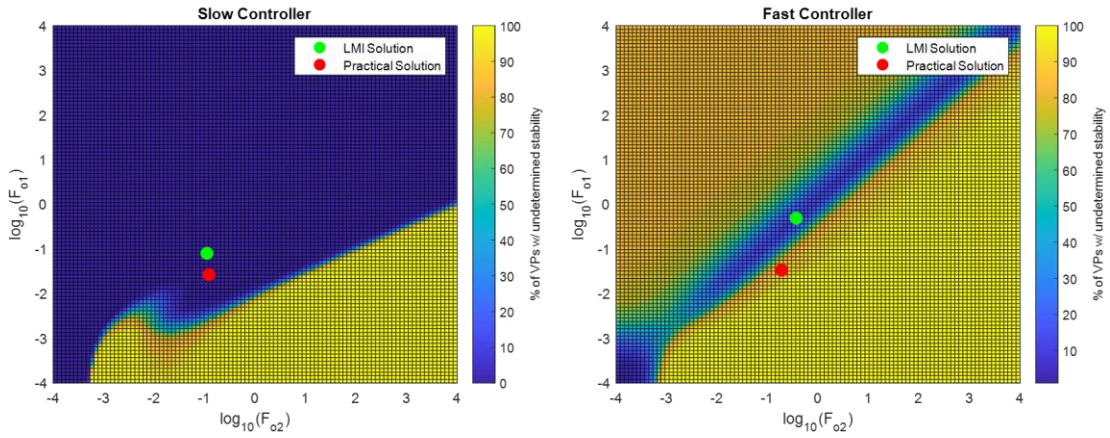

Figure 6-4: Color maps of observer gains and what percentage of SL model virtual patients were absolutely stable.

The green dots in figure 6-4 are the LMI solution observer gains. The red dots are the observer gains if one were to place the observer poles exactly 10 times faster than the controller poles, which is a common choice in observer design. For controller 1, the blindly chosen gain is absolutely stable, but for controller 2 it is undetermined (recall that the Lyapunov analysis used to assess absolute stability is ultimately a sufficient condition). In both cases the LMI design technique successfully finds the stable region.

The LMI design technique is at times necessary, depending on the problem parameters. For situations when the observer gain is not 2D and cannot be visualized as easily as in figure 6-4, the LMI design technique would be critical to avoid possibly unstable gains.

6.4 Discussion

The observer-based integral controller designed for absolute stability performed well, particularly for the slower controller. The controller was accurate with minimal overshoot across not only a range of physiologies, but different initial states,

showcasing its situational flexibility. The faster controller lost about 6 minutes in rise time from controller 1, but had worse performance metrics as seen in table 6-1. This highlights the main limitation of the observer-based integral controller. Because the controller was designed about a nominal plant, it will have overshoot when evaluated across a range of virtual patients. Fast controllers will exacerbate this issue possibly making the controller useless. The overshoot of over 7 mmHg produced by controller 2 would not be accurate enough for permissive hypotension if DCR were to be used [11]. This limitation is common among all non-adaptive controllers. Possible solutions are to only use slower variants (such as controller 1), which may be good enough for some resuscitation scenarios, or to tune the controller further to minimize the fast controller overshoot.

Ideally, we would like to compare this controller's performance to others in the literature. However, the simulation scenario used in this work is different enough from the literature that the reported results are not immediate comparable. Further work to apply previously developed controllers to the simulation scenario described in this work is expected.

Lastly, there is the potential to further showcase the importance of absolutely stable control. If it can be demonstrated that controller systems that are not absolutely stable have unstable or poor performance, then the LMI design method becomes critical for this application. The current simulation setup does not show this for reasonable non-absolutely stable controllers (such as the 10x observer pole solution). This suggests the nonlinear behavior of K_P is not interesting enough to trigger instability. Further work

to introduce realistic disturbances is required to truly display the importance of absolutely stable control.

Chapter 7: Conclusion and Future Work

In this final chapter, the results detailed in this work are summarized. In particular, the algorithms' novelty and limitations are recapitulated. A short discussion on the future work stemming from the current state of development is included as well.

7.1 Conclusions for the Prediction Algorithm

The BPPA is an algorithm which predicts blood pressure for decision support systems. While it mainly revolves around using a common EKF technique to estimate states and parameters of a model, applying this technique over multiple stages in the hemorrhage resuscitation process to make accurate BP predictions is a novel contribution.

The BPPA was evaluated on a cohort of virtual patients. It was shown that the BPPA accurately predicts blood pressure over 5-, 10-, and 15-minute intervals across a range of physiologies, and the uncertainty bound it generates captures any error well.

Rebleeding was introduced to assess the BPPA's ability to reject disturbances. With unknown hemorrhage present the BPPA's accuracy reduced drastically, and failed to outperform a simple constant BP prediction. There is an opportunity to enhance the BPPA with disturbance rejection qualities. Further work towards this goal is expected to look at MMAE, which could let the BPPA account for uncontrolled blood loss when it is present.

As mentioned in section 3.1, the purpose of the BPPA is to work with a DSS, making the development of such a system the clear next step. Ideally, the SL model

would be used again to make predictions for how a future bolus would affect blood pressure. The issue that arises here is that to predict how MAP will respond to a bolus the K_P must be accurately known. It is very difficult to estimate K_P purely from MAP data. It is unlikely the SL model is a viable solution to predict blood pressure responses to a bolus. Machine learning models have been applied to this problem and shown promise. A deep neural network was able to accurately predict responses to a future bolus having been trained on data from virtual patients. Further work is required to reconcile the BPPA and a machine learning methodology.

7.2 Conclusions for the Absolutely Stable Controller

An observer-based integral controller was designed to regulate blood pressure for resuscitation procedures. A novel LMI design technique was used to develop the controller such that it was absolutely stable over a range of physiologies which were expressed a class of nonlinearities.

A cohort of virtual patients was used to assess the controller's performance. The controller successfully regulated blood pressure to a target setpoint across all the VPs, although there was a clearly inverse relationship between speed and performance. Slower controllers had low overshoot and minimal steady state error, while faster controllers had overshoot and steady state error values that were not good enough for resuscitation applications.

Future work for the observer-based integral controller will revolve around making the design process less conservative. Several factors caused the LMI design method to potentially exclude portions of the absolutely stable solution space. One of which was that the Lyapunov analysis used to develop the LMIs is inherently a

sufficient condition. If a controller does not satisfy the LMIs in equation 5.21, we can say nothing about whether it does or does not stabilize the system. Another factor, was that, in order to have a unique solution when solving the LMIs, parts of the P matrix had to be set to zero. This immediately removed some solutions to equation 5.24 that the LMI solver could find.

Addressing these issues will indirectly ameliorate some of the challenges brought up in section 6.4. We would like to tune our controller to minimize the overshoot and steady state error even at fast rise times. By reducing the conservativeness of the absolutely stable solution space, there is the potential to find controller gains that allow for higher speeds without as drastic a deterioration in performance as was shown in section 6.2. Making the design process less conservative will also help identify what physiologies and disturbances cause instability, allowing for us to identify scenarios that highlight the importance of absolutely stable control.

References

[1] L. C. Sheppard, “Computer control of the infusion of vasoactive drugs,” Ann Biomed Eng, vol. 8, no. 4, pp. 431–444, Jul. 1980, doi: 10.1007/BF02363444.

[2] B. P. Kovatchev, M. Breton, C. Dalla Man, and C. Cobelli, “In Silico Preclinical Trials: A Proof of Concept in Closed-Loop Control of Type 1 Diabetes,” J Diabetes Sci Technol, vol. 3, no. 1, pp. 44–55, Jan. 2009, doi: 10.1177/193229680900300106.

[3] S. Bibian, G. A. Dumont, and I. Black, “Closed-Loop Target-Controlled Infusion Systems: Stability and Performance Aspects,” *Military Medicine*, vol. 180, no. suppl_3, pp. 96–103, Mar. 2015, doi: 10.7205/MILMED-D-14-00380.

[4] D. E. Ghetmiri, M. J. Cohen, and A. A. Menezes, “Personalized modulation of coagulation factors using a thrombin dynamics model to treat trauma-induced coagulopathy,” *NPJ Systems Biology & Applications*, vol. 7, no. 1, pp. 1–17, Dec. 2021, doi: 10.1038/s41540-021-00202-9.

[5] B. J. Eastridge et al., “Died of wounds on the battlefield: causation and implications for improving combat casualty care,” J. Trauma, vol. 71, no. 1 Suppl, pp. S4-8, Jul. 2011, doi: 10.1097/TA.0b013e318221147b.

[6] B. J. Eastridge, J. B. Holcomb, and S. Shackelford, “Outcomes of traumatic hemorrhagic shock and the epidemiology of preventable death from injury,” Transfusion, vol. 59, no. S2, pp. 1423–1428, 2019, doi: 10.1111/trf.15161.

[7] B. Yapps et al., “Hypotension in ICU Patients Receiving Vasopressor Therapy,” Sci. Rep., vol. 7, no. 1, p. 8551, Aug. 2017, doi: 10.1038/s41598-017-08137-0.

[8] C. G. Ball, “Damage control resuscitation: history, theory and technique,” Can J Surg, vol. 57, no. 1, pp. 55–60, Feb. 2014, doi: 10.1503/cjs.020312.

[9] Bendjelid, K., and Romand, J.-A., 2003, “Fluid Responsiveness in Mechanically Ventilated Patients: A Review of Indices Used in Intensive Care,” Intensive Care Med., 29(3), pp. 352–360.

[10] J. C. Duchesne et al., “Damage control resuscitation: the new face of damage control,” Journal of Trauma and Acute Care Surgery, vol. 69, no. 4, pp. 976–990, 2010.

[11] J. L. Sondeen, V. G. Coppes, and J. B. Holcomb, “Blood pressure at which rebleeding occurs after resuscitation in swine with aortic injury,” J Trauma,

vol. 54, no. 5 Suppl, pp. S110-117, May 2003, doi: 10.1097/01.TA.0000047220.81795.3D.

[12] A. Hirshberg, D. B. Hoyt, and K. L. Mattox, “Timing of fluid resuscitation shapes the hemodynamic response to uncontrolled hemorrhage: analysis using dynamic modeling,” J Trauma, vol. 60, no. 6, pp. 1221–1227, Jun. 2006, doi: 10.1097/01.ta.0000220392.36865.fa.

[13] E. J. Snider, D. Berard, S. J. Vega, G. Avital, and E. N. Boice, “Evaluation of a Proportional–Integral–Derivative Controller for Hemorrhage Resuscitation Using a Hardware-in-Loop Test Platform,” Journal of Personalized Medicine, vol. 12, no. 979, p. 979, Jun. 2022, doi: 10.3390/jpm12060979.

[14] Bighamian, R., Kim, C.-S., Reisner, A. T., and Hahn, J.-O., 2016, “ClosedLoop Fluid Resuscitation Control Via Blood Volume Estimation,” ASME J. Dyn. Syst., Meas., Control, 138(11), p. 111005.

[15] M. Alsalti, A. Tivay, X. Jin, G. C. Kramer, and J.-O. Hahn, “Design and In Silico Evaluation of a Closed-Loop Hemorrhage Resuscitation Algorithm With Blood Pressure as Controlled Variable,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 144, no. 2, Oct. 2021, doi: 10.1115/1.4052312.

[16] A. Tivay, G. C. Kramer, and J.-O. Hahn, “Collective Variational Inference for Personalized and Generative Physiological Modeling: A Case Study on Hemorrhage Resuscitation,” IEEE Transactions on Biomedical Engineering, vol. 69, no. 2, pp. 666–677, Feb. 2022, doi: 10.1109/TBME.2021.3103141.

[17] Bighamian R, Reisner AT, and Hahn JO, “A Lumped-Parameter Subject-Specific Model of Blood Volume Response to Fluid Infusion.,” Frontiers in physiology, vol. 7, p. 390, 2016, doi: 10.3389/fphys.2016.00390.

[18] J. M. Dickson, X. Wang, A. E. St. John, E. B. Lim, S. A. Stern, and N. J. White, “Damage Control Resuscitation Supplemented with Vasopressin in a Severe Polytrauma Model with Traumatic Brain Injury and Uncontrolled Internal Hemorrhage,” Military Medicine, vol. 183, no. 9–10, pp. e460–e466, Sep. 2018, doi: 10.1093/milmed/usy001.

[19] S. Sarkka, *Bayesian Filtering and Smoothing*, Cambridge University Press, 2013.

[20] B. Alexander, M. Cannesson, T. Quill. Blood Pressure Monitoring, 2013, doi: 10.1016/B978-0-323-11237-6.00012-1.

[21] H. K. Khalil, *Nonlinear Systems*, Third edition. Upper Saddle River, New Jersey: Prentice Hall, 2002.