

Control of Arterial Hemoglobin Saturation in Premature Infants using H-infinity
Synthesis and Performance Specifications from Best Clinical Practice

A Thesis presented to the Faculty of the Graduate School
University of Missouri – Columbia

In Partial Fulfillment
of the Requirements of the Degree
Master of Science

By
DANIEL QUIGLEY
Dr. Roger Fales, Advisor
July 2013

The undersigned, appointed by the Dean of the Graduate School, have examined the thesis entitled

CONTROL OF ARTERIAL HEMOGLOBIN SATURATION IN PREMATURE
INFANTS USING H-INFINITY SYNTHESIS AND PERFORMANCE
SPECIFICATIONS FROM BEST CLINICAL PRACTICE

Presented by Daniel Quigley

A candidate for the degree of Master of Science

And hereby certify that in their opinion it is worthy of acceptance.

Professor Roger Fales

Professor Craig Kluever

Professor Carmen Chicone

Acknowledgments

I would like to thank my supervisor, Dr. Roger Fales, whose technical expertise and experience with the subject was priceless in guiding me through the research process. Also for his flexibility, patience, and genuine caring attitude that made this thesis project personal and fun. I would like to thank my family, friends, and teammates who have always helped push me to pursue excellence in everything I do. A special thanks to Shannon Leinert for being my support team as we worked together to finish our degrees on time. Finally I would like to thank my committee members Dr. Craig Kluever and Dr. Carmen Chicone for their time.

TABLE OF CONTENTS

Acknowledgments	ii
List of Figures	iv
List of Tables	v
ABSTRACT.....	vi
Chapter 1: Introduction	1
1.1 Background.....	1
1.2 Literature Review	4
1.2.1 Theoretical work.....	4
1.2.2 Prior Clinical Testing of similar devices	5
1.2.3 Prior work at the University of Missouri.....	9
1.3 Objectives and motivation	12
Chapter 2: Model	16
2.1 Considered Models	16
2.2 Three Transfer function model.....	18
2.3 Genetic Algorithm	22
2.4 Non-linear model	28
Chapter 3: Control design.....	30
3.1 Performance Specifications	30
3.2 H-infinity synthesis.....	34
3.3 Performance Analysis	36
Chapter 4: Results.....	39
4.1 Controller Synthesis results.....	39
4.2 Open loop parameter estimation	42
4.2.1 Constant parameter estimation.....	43
4.2.2 Changing parameter estimation	47
4.2.3 Parameter estimation with real data	52
4.3 Closed loop simulations	55
Chapter 5: Prototype design and testing.....	59
5.1 Device Performance Specifications	60
5.2 Drawings and Final Design.....	62
Chapter 6: Discussion and Conclusion.....	65
6.1 Discussion	65
6.1.1 Open loop controller selection	65
6.1.2 Closed loop controller selection	68
6.2 Recommendations.....	69
6.3 Conclusion	70
Appendix A: Model information	72
Appendix B: Controller Results.....	73
Appendix C: Prototype Drawings.....	75
Appendix D: Genetic Algorithm MATLAB code	79

List of Figures

Fig. 1 Data collection setup [23]	12
Fig. 2 Three transfer function model of infant response	19
Fig. 3. Multiplicative uncertainty for nominal plant 1	21
Fig. 4. Model block diagram with multiplicative uncertainty	22
Fig. 5. GA window example with estimated model shown to match the feedback information.....	23
Fig. 6. Flowchart of the parameter estimating genetic algorithm model	27
Fig. 7. Parameter estimation simulations with closed loop FiO_2 control.....	28
Fig. 8. Block diagram of non-linear model of $\text{FiO}_2 - \text{SpO}_2$ relationship.....	29
Fig. 9 Block Diagram of model, performance weights, and uncertainty model.....	34
Fig. 10. Closed loop step response of controller without rate limiting performance specification.....	40
Fig. 11. Response of a controller synthesized using the rate limiting specification	41
Fig. 12. Controller 1 parameter estimation simulation	44
Fig. 13. Error used to assign fitness of parameter sets during simulations	45
Fig. 14. Yellow line represents the correct choice, red is simulation result	46
Fig. 15. Result when the correct parameters are between controllers.....	47
Fig. 16. Estimating varying time constants test.....	48
Fig. 17. Controller selection for varying time constant test.....	48
Fig. 18. Estimating varying time constants test 2	50
Fig. 19. Controller selection, varying parameters test 2	50
Fig. 20. Changing gain parameter estimation	51
Fig. 21 Changing gain controller selection	52
Fig. 22. Raw data simulation with unknown model parameters.....	53
Fig. 23. Raw data simulation controller selection	54
Fig. 24. Input data from clinical setting used in GA testing	54
Fig. 25. Closed loop parameter estimation.....	56
Fig. 26. Closed loop control test – SpO_2 and control effort.....	56
Fig. 27. Modeled closed loop results with real disturbance inputs from data	57
Fig. 28. Real disturbance input data used	58
Fig. 29. NEO2 Blend valve retrofit design assembly drawing.	59
Fig. 30. Control panel and hardware box	62
Fig. 31. Original prototype design, circa 2011	64
Fig. 32. New prototype design completed.....	64
Fig. 33. Bode magnitude response of controller K for nominal plant 2	74

List of Tables

Table 1 Parameter uncertainty ranges for FiO_2 transfer function.....	19
Table 2 Parameter ranges for Heart Rate and Respiratory Rate	19
Table 3 Performance and Stability verification values	39
Table 4 Uncertainty Model (W_A) for each of the six plants.....	72
Table 5 Uncertainty Model for Heart Rate and Respiratory Rate	72
Table 6 Results of controller synthesis	73

ABSTRACT

This work details the development of a novel control system for regulating the Arterial Hemoglobin saturation (SpO_2) of premature infants. An H-infinity synthesis method was used to derive multiple controllers to cover a large range of parametric uncertainty in the model. Performance specifications were defined using manual control protocols, which are clinically proven to reduce the rate of Retinopathy of Prematurity (ROP) in the target patients. The controllers were tested using previously validated models and shown to meet the robust stability and performance specifications. In order to switch between controllers and adapt to the changing physiology of the patients a genetic algorithm was modified and tested. This algorithm was proven to adequately track changing model parameters and select an appropriate robust controller for providing closed loop control. In addition a physical prototype was designed and built to facilitate automatic control of FiO_2 during clinical tests of the control algorithms.

Chapter 1: Introduction

1.1 Background

Respiratory distress syndrome (RDS) is a breathing disorder in preterm infants, which is caused by underdeveloped lungs and insufficient surfactant production[1]. This syndrome is primarily seen in premature infants with the youngest gestational age being the most susceptible but a very small percentage suffer from a genetic inability to produce surfactant. The lack of surfactant in the alveoli inhibits the lung's ability to properly inflate, which is necessary for ventilation. RDS is the leading cause of death for infants less than 31 weeks gestational age[2].

Oxygen enriched air, with the level of oxygen described as the fraction of inspired oxygen (FiO_2), is supplied to the infant and has been a standard protocol to combat RDS since the early 1940's[3]. This oxygen therapy is used with respiratory support devices like a nasal cannula which is a simple tube that delivers oxygen to the nostrils, ventilator which mechanically moves air in and out of the lungs, CPAP which provides continuous positive airway pressure but the patient breaths on their own, or other respiratory support modalities. Despite the use of artificial surfactant therapy, the need for supplemental oxygen persists.

There are many health risks associated with supplemental oxygen, however. High blood oxygen concentrations can cause retinopathy of prematurity (ROP). ROP is a disease of the eye in which the vascularity of the eyeball grows in a rapid and disorganized manner. Preterm infants are susceptible because the eyeball develops late in the gestation period. The rapid growth of the retinal blood vessels is in part incited by episodes of hyperoxia, which can result in retinal scarring and possible blindness. This is

an obvious drawback to supplemental oxygen therapy for premature infants. While some cases of ROP naturally correct themselves, many need surgery to regain sight for the infant. In extreme cases total blindness can occur such as the case of famous musician and entertainer Stevie Wonder[4].

The risk of ROP is associated with high arterial hemoglobin saturation (SpO_2), but low SpO_2 concentrations can result in hypoxia and tissue damage[5]. For these reasons it is very important to maintain a safe SpO_2 range for these at risk patients. This is done in hospitals across the world by nurses and doctors using a manually operated analog gas mixing device. Typically digital sensors provide feedback on the FiO_2 , SpO_2 , heart rate (HR), and respiratory rate (RR).

The level of FiO_2 supplied to the infant is based on nurses' judgment and protocols for adjusting it are typically mandated by the individual medical doctors at the hospital. Arterial oxygen saturation is most commonly monitored by pulse oximetry. This is a method which improved upon the more invasive indwelling umbilical catheter electrode by using a laser to determine the SpO_2 . The laser detects the red color of the blood while flowing through extremities and determines the saturation of oxygen by the richness of this red color. The current techniques used allow for fast sampling and averaging algorithms have been able to give accuracy in the range of $\pm 2\%$ for saturations above 70% [6].

The desired SpO_2 range as agreed upon by the literature has fluctuated since the early studies but the most current research shows 85% to 93% is the most effective range for avoiding ROP and hypoxia [7]. There are alarms present in the neonatal intensive care unit (NICU) that notify medical attendants of a dangerous SpO_2 . However, it has

been shown that preterm infants spend approximately half of the time was spent within the clinically intended range of SpO₂. More specifically 20% of the time was spend below the range and 30% of the time above this range during routine neonatal intensive care [8, 9]. One study even revealed that the upper limit alarms had been set too high 76% of the time because of the frequency that they were exceeded[10]. These statistics indicate that there is still a large potential for improvement of the time spent in a healthy range despite advancements in technology and knowledge of the subject. A small number of researchers have been working on using closed loop automatic control techniques to do this since the late 1970's; Beddis et al, Dugdale et al, Bhutani et al, Morozoff et al, Claur et al, and Urschitz et al. An overview of these works with their significance is provided.

One of the most challenging aspects of applying automatic control is the changing physiology of the patients. There is a wide range of gestational ages, birth weights, and varying degrees of health that affect the response of each infant to FiO₂ adjustments. In addition, as each infant grows the changing physiology represents the need for a controller that can recognize and adapt to the varying parameters or one that is robust enough to provide effective control for all parameters. The literature shows researchers have chosen one these two options but there is potential for a robust controller that is also adaptive.

1.2 Literature Review

1.2.1 Theoretical work

Several control systems for the ventilation of premature infants have been presented in literature. Tehrani et al. proposed using a proportional, integral, derivative (PID) controller with feedback arterial partial pressure of oxygen in newborn infants in 1991 [11]. The performance of the control system was investigated using computer simulation. The results under two different test conditions indicate stable performance for each case. Many of the other controllers in literature include adaptive components. Yu et al. developed a multiple-model adaptive controller (MMAC) to regulate oxygen saturation by changing FiO_2 in 1987 [12]. The MMAC procedure assumes that the system can be represented by one of a finite number of models. A controller is designed for each of these models that have constant parameters. The actual control output is a weighted sum of each individual controller. The MMAC was found to effectively regulate oxygen saturation in animal studies. Taube et al. developed an adaptive PID control system based on the model by Grodins that was then simplified by Sano and Kikucki to a static relationship in 1991 [9, 15, 16]. The adaptive part of the controller involves updating the KL gain by a ratio of the partial pressure of arterial oxygen and the input FiO_2 . They found that the automatic regulation of FiO_2 may be a better method than manual adjustments.

1.2.2 Prior Clinical Testing of similar devices

A history of past clinical trials was used to more fully understand the progression of the available computing technology, control algorithm research, and understanding of the patient's needs. A brief synopsis of these clinical trials follows.

The first published study of closed-loop automatic control of inspired oxygen was done in 1979 using arterial oxygen tension (PaO_2) feedback on a 1 minute interval from an indwelling umbilical electrode[13]. The FiO_2 was controlled in discrete steps of 5% and used with intermittent positive pressure ventilation, continuous positive airway pressure, and hood support. Limits for PaO_2 were set between 7.3 kPa and 10.7 kPa. The amount of time spent outside these limits decreased from 27.6% to 12.2% with use of servo control compared to manual. The mean duration of the episodes below the limits decreased from 6.1 to 2.5 minutes and mean duration above decreased from 4.7 to 1.7 minutes. However, the number of episodes above the set limits actually increased during servo control. This could have been an issue with the sensor feedback or a problem inherent in the control system, which indicates there is room for improvement of either control algorithm or equipment capability.

The second clinical study involving infants was done in 1985 and also utilized an indwelling umbilical electrode[14]. The target value for PaO_2 was chosen to be 10 kPa measured every minute, and FiO_2 was controlled in steps of 5 or 10%. The authors considered adaptive control options for this application, but decided on a simple controller based on an estimated steady state gain and time constant of the response to a step input. This controller was able to tolerate up to 300% variation in static gain parameters. The researchers chose to avoid adaptive control because non-linear problems

are difficult to analyze and adaptive controllers are more complicated than conventional options. Control parameters were chosen conservatively (1 kPa per % O₂ gain and a 120 second time constant) to ensure stability over a wide range of operating conditions because quality of control was only a secondary goal. The mean time spent within 1kPa of the set point was 74.9% with automatic control, and 45.2% without automatic control. The authors proposed that an adaptive controller could be developed in the future with the goal of improving care. At the time of the study, a parameter-estimating algorithm was suggested, but the required level of engineering sophistication for this was not available.

By 1993 pulse oximeter technology had developed enough that measuring the blood oxygen saturation (SaO₂) with non-invasive devices was possible. The first automatic control study to be done using this technology was published by Morozoff et al. They used the feedback to define discrete states that described the direction of the error, velocity, and acceleration while ignoring the magnitude of those values. The states merely describe whether the SaO₂ is above or below the setpoint, and if the SaO₂ is trending toward or away from the set point. FiO₂ is then either increased or decreased relative to its previous value. This controller required very little processing power and was adept at controlling the SaO₂ except when helping recover from sudden desaturation events. In these instances manual intervention was required. The results showed that the acceptable range of 90-95% SaO₂ was maintained 50% of the time during automatic control while only 39% of the time under manual control[15].

A study done in 2001 is the first to use a novel closed loop controller in comparison to a fully dedicated nurse performing continuous monitoring and adjustments

as opposed to routine monitoring [16]. This study used SpO₂ feedback and a set of rules to adjust the FiO₂ when outside the range of 88-96%. These rules covered cases including slow changes during apparent stability and rapid cases of desaturation. The mean weight and gestational age was 712g and 25 weeks. The SpO₂ feedback period was chosen to be 2 seconds but FiO₂ adjustments were allowed to be continuous. Analysis showed that manual and automatic control resulted in similar frequency and mean duration periods of hypoxemia, but under automatic control the number of overshoots to hyperoxemic ranges was reduced from 5.8±7.8% to 1.5±1.9%. The percent of time spent in the goal range of 88%-96% was improved from 66.3±13.9% to 74.9±12.5%. This study is the first to truly prove the efficacy of closed loop control as a replacement for a fully dedicated nurse or regularly attending nurse. This control system doesn't require manual intervention as did the controller in the study by Morozoff and Evans, and proves a savings in nursing time since normal conditions require nurses to perform other tasks in addition to constantly monitoring SpO₂.

Urschitz et al. furthered the research in 2004 with another clinical trial testing a new style of controller. This controller is based on artificial intelligence research and a method called temporal-abstraction, which helps filter the feedback and reduce the number of adjustments the controller must make [17]. The controller was programmed by monitoring clinician FiO₂ adjustments but, similarly to previous work, was not able to respond to acute severe hypoxic episodes since it had a built-in wait period of 180 seconds after each adjustment. It was, however, able to increase the amount of time SpO₂ was maintained in an acceptable range by 10-15% [18]. Urschitz et al. also made reference to the need for more research to determine an optimal range for SpO₂, as well

as the need for further work on improving the sensor technologies to avoid false feedback.

The team of Morozoff and Smyth published results testing three different controller options, including the first adaptive controller, in 2009. The first controller was called a State Machine Algorithm that uses the error, error velocity and error acceleration as inputs to a set of rules that determine the FiO_2 change similar to their previous controller tested in 1993. The second controller tested was a classic PID style controller with gains that could be changed “on the fly” by user input. The adaptive controller uses a linearized model of the oxygen dissociation curve and either increases or decreases the slope of certain linear segments as dictated by the SpO_2 feedback [19]. The adaptive model proved best and had the smallest need for manual intervention; in second was the state machine, and finally the PID. All three controllers improved upon manual control[20]. The significant advancement from this work is the effective use of an adaptive controller that is capable of changing parameters with changing neonate physiology.

Claire et al. also pursued new research in this area and published another clinical study in 2009. They used an oximeter with an eight second feedback interval and allowed the control system to adjust FiO_2 every second. The specifics on the controller used were not published, but it considers an averaged basal FiO_2 as a moving baseline and makes adjustments based on that when the SpO_2 is out of the intended range. A safety measure that was introduced would select the median FiO_2 from the past 60 seconds if the feedback signal was lost for more than 10 seconds. They were able to show that the automatic control increased the time the infants spent in the range of 88-

95% from $42\% \pm 9\%$ to $58\% \pm 10\%$ [21]. It also showed an increased time the infants spent at lower FiO_2 settings, which is thought to help avoid ROP. There were, however, an increased number of episodes below the desired range, which do not correspond with oscillations or hyperoxemic events, suggesting that fine-tuning of the controller is possible.

Claire et al. continued their work with a much larger study in 2011 with 32 infants at four different hospitals using the same automatic controller as before. They recorded two consecutive 24-hour periods, one with manual FiO_2 control and the other with automatic control. This time a target range of 87%-93% was used and again improvements over manual control were seen. Also, the significant increase in time spent below the desired range was seen as before[22]. The authors note that the infants chosen to participate in the study were those with the most frequent fluctuations in SpO_2 and therefore they possessed the greatest risk for ophthalmic or neurologic injury.

1.2.3 Prior work at the University of Missouri

Two previous graduate students at the University of Missouri have conducted prior non-clinical published investigations of the device. Bradley Krone, a graduate student who has since received his M.S. for his work on this project, analyzed the relationship between FiO_2 and SpO_2 and found that it can be modeled using a first order transfer function. Krone concluded that “through the analysis of the biological system and recorded data it is thought that by considering the relationship between the SpO_2 and the FiO_2 , HR, and RR that a model will be able to better simulate the SpO_2 response to inputs” [23]. Several different transfer function models were tested, and the dynamic

transfer function model was found to model the SpO₂ accurately for a prolonged amount of time. On average, the function is able to model SpO₂ adequately for 15 minutes or more. Furthermore, the model is “reliable over many situations and can adequately estimate the gains and time constants in simulated data” [12]. Thus, the controller operates using a model that accurately and reliably simulates real-life data.

Next, tests were performed in order to determine the best controller to use. The test compared the parameters of maximum SpO₂, maximum FiO₂, average SpO₂, and average FiO₂. Three different controller types were tested for a simulation with constantly changing FiO₂, HR and RR to better simulate the changing parameters of actual infants. These controllers included a linear quadratic regulator proportional integral controller (LQR-PI), a robust controller designed using H-infinity optimization, and an adaptive controller with feed forward disturbance rejection (AC-FFDR) [23]. The results showed that “all controllers attempted to reject the disturbances caused by variations in HR and RR and keep the SpO₂ at a given set point,” and that the robust controller has the best performance because it “has the lowest maximum SpO₂ and average SpO₂.” While the robust controller had the largest maximum FiO₂ and average FiO₂ values, these are well within the limitations of the FiO₂[23].

Keim, a former student at the University of Missouri who has earned his Ph.D. for his work on this project, further addressed the effectiveness of the device controls in his dissertation “Control of Arterial Oxygen Saturation in Premature Infants” [24]. To further investigate the controller’s performance, actual infant response data was collected during a study performed at Columbia Regional Hospital, now known as University Women’s and Children’s Hospital. The patients involved were premature infants with frequent

desaturation periods. Clinical data was collected for patients on ventilators, FlowPAP, and SiPAP respiratory support devices which are both specialized devices for positive airway pressure. A LabView program collected heart rate, respiratory rate, SpO₂ and FiO₂ signals. The oxygen sensor transmitted data every second, and the heart rate, respiratory rate, and SpO₂ were collected every 5 seconds. The pulse oximeter recorded these values. The data collection program runs continuously saving one-hour long files of the signals. To more completely capture the effects of surrounding circumstances, nurses kept diary files of the patient care-giving information. Such information included gestation age, adjusted age, respiratory modality, oxygen alarm parameters, and current medications.

Three different estimation systems were evaluated, including a dynamic fuzzy logic system, a continuous parameter-estimating extended Kalman filter, and a discrete parameter-estimating extended Kalman filter [24]. An adaptive controller was developed based on the discrete extended state observer and parameter-estimating extended Kalman filter system. The adaptive controller was found to display the best performance for a system with unknown disturbances and large ranges of model parameters when compared to a static proportional integral and a robust controller. It displayed the fastest and most accurate performance in the disturbance rejection simulation. First, the microcontroller receives the patient's SpO₂, heart rate, respiratory rate, and percentage of oxygen through serial cables. The adaptive control system then uses these signals to determine the optimal oxygen percentage to administer to the infant. It will adjust FiO₂ levels and also isolate the apnea alarm sent from the bedside monitor, using it to activate the flow control motor and vibration motor [24]. Keim concludes that the "software is shown to run in real

time in less than 5 seconds.” He continues saying that “this prototype should be capable of administering the same level of patient care as a nurse.” In addition to being at least as good as a nurse, the “prototype should aid the nurse in providing a more consistent and higher level of care for the premature infants in the NICU,” [24].

These two researchers, Krone and Keim, based their work on data they collected in a clinical setting from target patients. The hospital sensors were used to collect feedback information on a laptop. A representation of the sensor setup and data collection procedure can be seen in Fig. 1. The data they collected was analyzed and used to justify the models they created. This data was also used to verify the model and controller that is presented in this work.

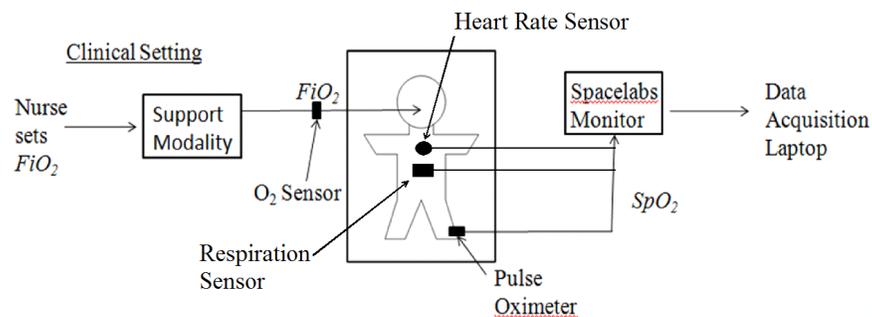


Fig. 1 Data collection setup[23]

1.3 Objectives and motivation

As discussed above a recent study testing strict guidelines for increasing and decreasing, or weaning FiO_2 and monitoring O_2 saturation manually, the researchers showed a significant reduction in rates of ROP [1]. Those results were repeated in a larger study and helped prove that proper manual control can be very effective in reducing the incidence of ROP [25]. As these advances in knowledge about the physiology and health of the patients continue to improve there are new opportunities for

closed loop control to match the success of the medical research. Better understanding of the patient's needs can be coupled with improved computing options and control theory to achieve new standards of care.

Multiple clinical studies have proven the efficacy of an automatic control algorithm for managing the FiO_2 compared to a nurse's control [13-16, 18, 20-22]. A few of these studies have shown a significant decrease in ROP rates and represent a reduced workload for the attending nurse if employed full time. Many different control strategies have been applied from simple proportional integral derivative control to fuzzy logic, a state machine rule set, and even an adaptive model based on the O_2 dissociation curve [19].

There is a noticeable lack of published work in that draws on both realms of research. Under strict monitoring and procedures manual control of FiO_2 can be very effective and automatic control is proven to be as effective or better than manual control in some cases. The need to take a medically justified approach to these control problems is apparent.

Here we attempt to utilize the best manual control methods and draw on previous control work to synthesize a controller to replicate those results. The goal is to use the best manual control practices as a guideline for defining how the robust controller operates and therefore improve automatic control even further. The best adaptive control methods are difficult to use with non-linear systems. Using adaptive parameter estimation techniques is still a viable option but they may not be accurate enough to rely solely on their parameter estimation. Therefore a genetic algorithm method that is continuously adapting to sensor feedback is used to select between the robust controllers.

Since each robust controller is capable of closed loop stability under a range of uncertain parameters they only require a “close enough” estimation of the model parameters.

Using this method incorporates the best of both worlds. The robust controllers will provide precise control with the safety of operating on a large range of uncertain parameters but the adaptive genetic algorithm has the flexibility to select a more appropriate controller as the patient’s physiology changes.

To be very clear the application of either robust control or adaptive control methods to this problem is not new or novel. This research intends to contribute the following to the literature:

- Adaptation of clinically tested manual control procedures into automatic control performance specifications
- Explore the use of a rate limiting performance specification to limit fast responses of the control effort
- Improve a genetic algorithm technique for real-time model parameter estimation
- Combine the adaptive model estimator with a robust control system in an intuitive and seamless way to provide the best possible patient care

Chapter 2 discusses modeling of the premature infant’s respiratory system and its response to different inputs. The genetic algorithm technique for model parameter estimation is also described. In Chapter 3 the controller synthesis is described with definitions of the performance specifications. The results of the parameter estimation tests and closed loop controller results are given in Chapter 4. The prototype that was

designed to carry out future clinical testing is illustrated in Chapter 5. Discussion of the results, limitations of the proposed design, and conclusion is given in Chapter 6.

Chapter 2: Model

2.1 Considered Models

The modeling of a premature infant's SpO_2 is a difficult task. This is partly due to the wide range of birth weights and gestational ages seen in the NICU setting that represent very different levels of development. It is not easy to tailor a model for each infant and also have the ability to tune that model as the infant's health improves or degrades. The models attempt to do this by describing the relationship between the inputs and outputs as defined by the model. It has been shown that not only the FiO_2 has an effect on the patient's SpO_2 but also the HR and RR which are all easily monitored and very useful for modeling purposes.

However, the problem is complicated further by changes in the infant's physiology that cannot be monitored or accounted for. An example of this is called shunting, which is when some amounts of venous blood bypasses the lungs and therefore is not transporting Oxygen to the body[26]. Sometimes this is caused by the alveoli filling with fluid which causes parts of the lung to be unventilated [27]. Circumstances such as this are difficult to monitor without invasive measures so they are difficult to account for. When such circumstances exist there is a change in the output (SpO_2) but no corresponding change in the measurable input (FiO_2 , HR, RR).

There are a few published models in this field, each with its own assumptions and limitations. Some are merely descriptive models that only attempt to map the relationship between inputs and outputs. Others are constitutive models that use equations trying to approximate the actual physical processes of the system. Keim adopted such a model which he adapted into a non-linear relationship from a linear

respiratory system model proposed by Yu [28]. These models use an equation that describes the dissociation of oxygen into the blood, which is based on partial pressure differentials of the gases, pH, temperature, and other chemical reactions involved with respiration. This model is based on many complex physical relationships and uses safe assumptions to distill the important aspects. Since it is based on real physical interactions it is highly tunable and can be very accurate.

The drawback to using such a model is that most of the tunable aspects of the model are not measureable and must be given assumed constants. It also only relates SpO_2 to FiO_2 but other measureable inputs have an effect on SpO_2 . The complexity of the model, its non-linearity and required computing power, are all drawbacks. For these reasons it wasn't selected as the main model for this research. It was, however, selected as a secondary means of verification of the models and controllers in this work as described later.

Other options include the neural network model and fuzzy logic model developed in the work by Krone [23]. These two options both attempted to utilize all available feedback information to help define the model. They have the added benefit of being dynamic models. That is to say they are able to analyze sets of data and adapt the model to the data given. This is a benefit because of the desire to achieve an adaptive controller. These models were not able to achieve a level of accuracy that was sustainable over a useful length of time.

The model ultimately selected was the three transfer function model that was developed by Krone. This was chosen for its use of all three measureable inputs, simplicity, accuracy in describing clinical data, and computational speed. The other

reason is for its close tie in to the genetic algorithm approach of parameter estimation for creating adaptive control. The genetic algorithm is discussed in detail in Chapter 3.

2.2 Three Transfer function model

The model used to develop the controller in this work is a combination of three first order transfer functions. It uses one transfer function to describe the relationship between FiO_2 and SpO_2 . The other two describe the relationship between the HR and RR to the output SpO_2 and are treated as disturbance inputs[23]. The work by Krone and Keim confirmed that the infant's response could be modeled as a first order transfer function approximation [23, 24]. These transfer functions are given in the form

$$\frac{G_x}{T_x + 1} \quad (1)$$

where G and T are the gain and time constant associated with the input x which can be HR, RR, or FiO_2 . The output of each function is SpO_2 . The parameters for these transfer functions were defined by clinically collected data from premature infants [12]. The FiO_2 transfer function represents the relationship between the inspired oxygen and the O_2 saturation while the HR and RR functions are considered disturbance inputs which also effect the SpO_2 . The model used can be seen as a block diagram in Fig. 2.

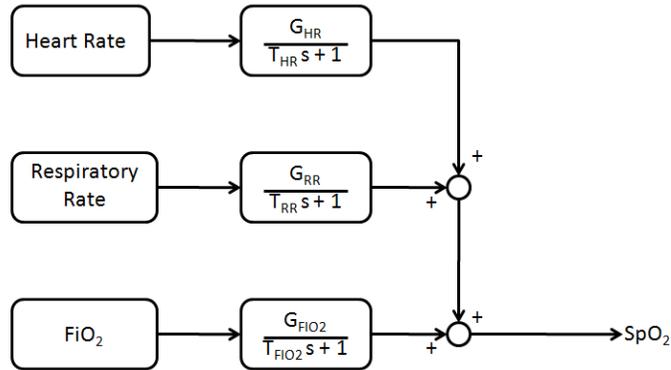


Fig. 2 Three transfer function model of infant response

The gains and time constants that are used in this model were defined by data collected in a NICU [12, 23]. By using different sets or different combinations of these six parameters it is possible for the model to very closely match the clinical data. Estimating the best combination of parameters that most accurately describe the particular patient that it is being applied to is what is referred to as adaptive modeling. Using that information for closed loop control is considered adaptive control.

Table 1 Parameter uncertainty ranges for FiO₂ transfer function

Plant #	Gain	Time Constant
1	0.25 - 3	60-130
2	0.25 - 3	110-180
3	0.25 - 3	160-220
4	3 - 8	60-130
5	3 - 8	110-180
6	3 - 8	160-220

Table 2 Parameter ranges for Heart Rate and Respiratory Rate

HR		RR	
Gain	Time Constant (sec)	Gain	Time Constant (sec)
-1.35 - 1.2	101 - 253	-1.4 - 1.3	68 - 234

The entire range of uncertain parameters is broken down into six different nominal plants. This is because it is not feasible for one controller to achieve robust stability for such a large uncertainty model. The ranges of FiO_2 parameters are shown in Table 1. The ranges of HR and RR parameters are the same for each of the 6 controllers and are shown in Table 2. For both tables the nominal value used for the model is the mean of the range displayed.

These ranges are used to construct uncertainty models that help define and verify the performance of the controller. As seen in Table 1 time constants of the FiO_2 ranges overlap. This is a safety net that has been designed to ease selection of which nominal plant best describes the patient. In this way a model with a time constant between two nominal plants will be accurately represented by either plant.

A transfer function uncertainty model that bounds the maximum of the frequency responses of the perturbed plant describes the parametric uncertainty of the nominal plants. This model is achieved by analyzing the frequency response of the multiplicative error of each perturbed plant with respect to the nominal plant then finding one transfer function that bounds the maximum of these errors. This multiplicative uncertainty is more clearly described as the equation in frequency domain

$$l(\omega) = \max \left| \frac{G_p(j\omega) - G(j\omega)}{G(j\omega)} \right| \quad (2)$$

where G is the nominal plant and G_p is the perturbed plant. The uncertainty models shown as $W_{\text{HR}}W_{\text{RR}}$ and W_{FiO_2} in Fig. 4. will bound the magnitude of the frequency response of equation (2). An example of the possible perturbed plants and the bounding equation for nominal plant number 1 can be seen in Fig. 3. The red line is the

uncertainty model bounding all possible plants. The equation for this uncertainty model can be found along with the uncertainty for each model in Appendix A.

These uncertainty models of each transfer function are used to help define the uncertainty of the entire model as in Fig. 4. where Δ is a stable and proper transfer function with H-infinity norm less than 1. In this way all possible plants are described and can be analyzed during controller synthesis.

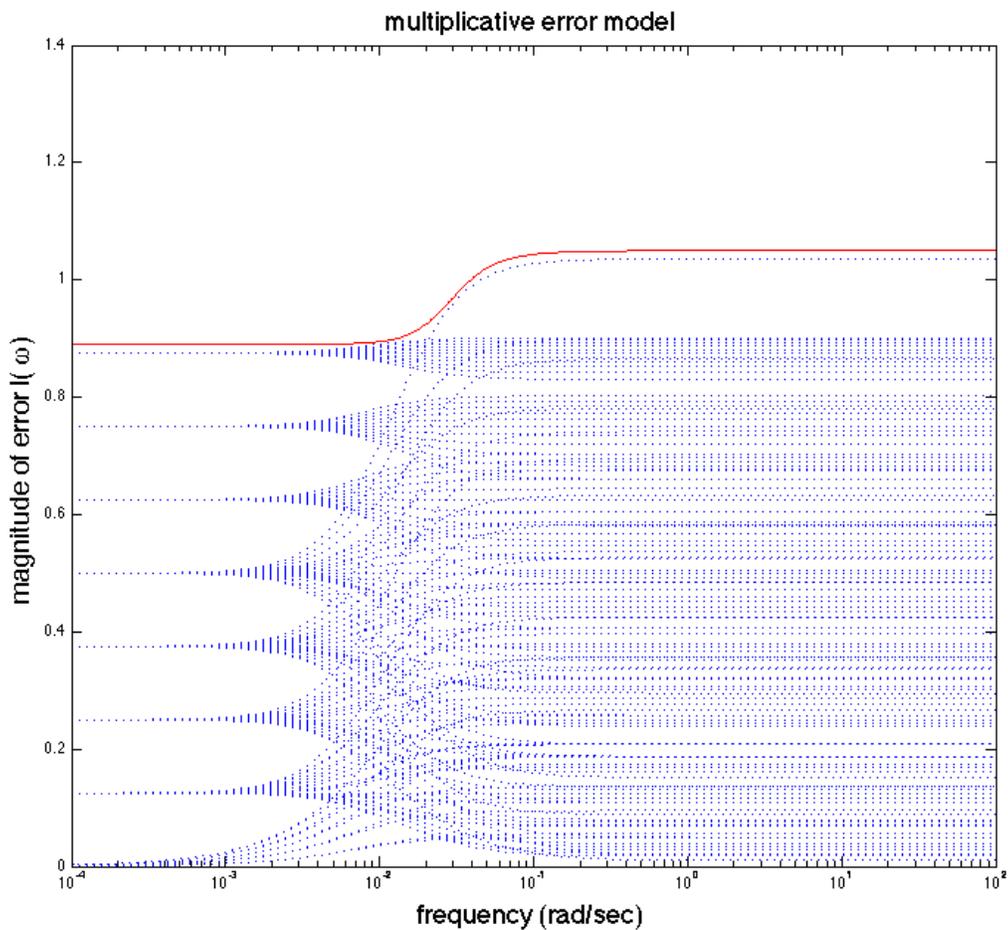


Fig. 3. Multiplicative uncertainty for nominal plant 1

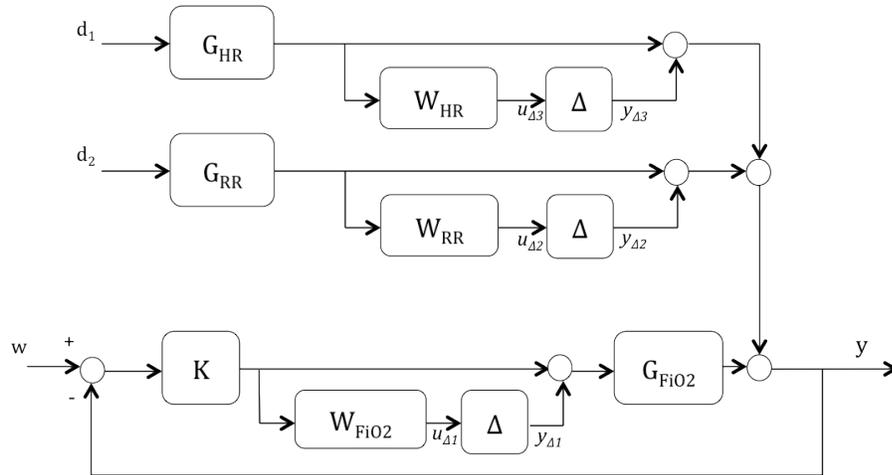


Fig. 4. Model block diagram with multiplicative uncertainty

2.3 Genetic Algorithm

The genetic algorithm is a selection method used to determine which nominal plant is currently most accurate at describing the patient's response. It takes a 5 minute window of feedback information containing the three inputs and the one output and estimates which set of six parameters best describes the response. This window must be large enough to properly analyze the dynamics of the transfer function responses to inputs, but also short enough to be able to adapt to changing parameters and reduce processor time. An example of a feedback window with inputs, a simulated SpO₂, and the estimated output "best guess" according to the GA is shown in Fig. 5. In this example a set of model parameters and real recorded inputs were used to obtain SpO₂ data that was sent to the GA to verify its ability to estimate the correct parameters.

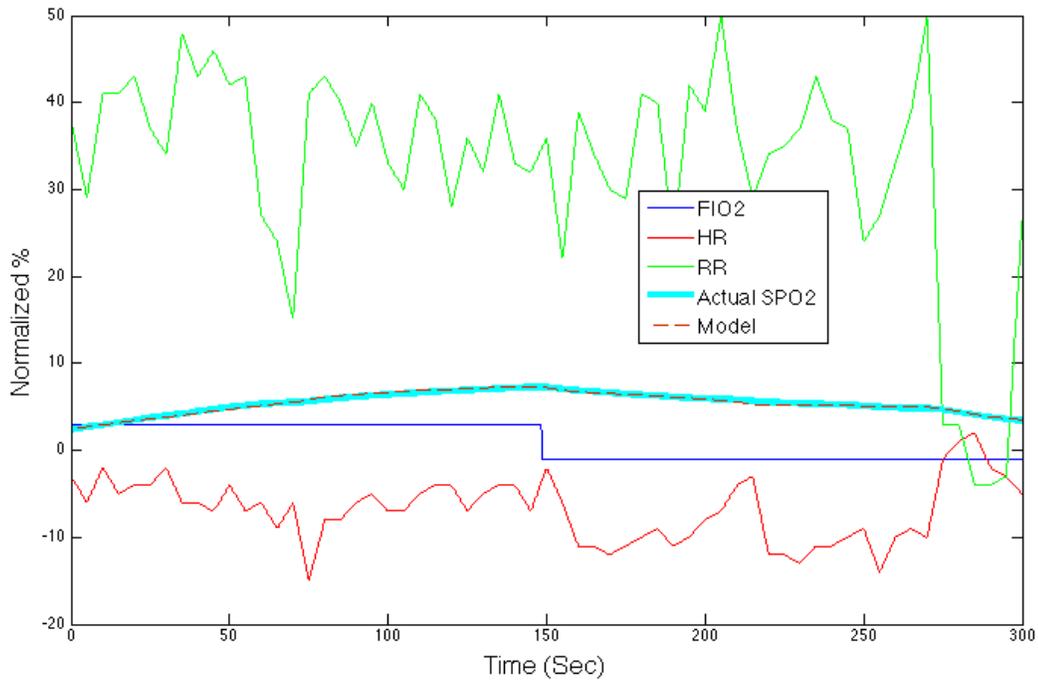


Fig. 5. GA window example with estimated model shown to match the feedback information

The original method for using a genetic algorithm in this application was developed by Krone and is improved in this work. The estimator is sent a window of data and an array of parameters, or *alleles*, which are initially chosen randomly from the prescribed range. These alleles are used to create a set of models with the feedback data as input. This is done in the MATLAB® environment using the `lsim` command and the transfer functions are represented as state space models to more easily define the initial conditions. The outputs of each model are then compared to the actual SpO₂ feedback data. Their relative “fitness” is described as the mean square error. The parameter set of the model with the lowest error is stored and passed on to the next generation as in evolution. The MATLAB code that is used to execute this method is shown in Appendix D.

The remaining sets of parameters are sent to a series of programs that mate certain sets of parameters and randomly mutate some parameters at a specified rate. This maintains a diversity of options, helps avoid local minimums, and also helps the system to converge on the best set of parameters. Local minimums are defined as set of parameters that are not the best descriptors of the model but changing those parameters in the right direction initially creates a larger error. The new group of alleles is then passed back to the calling function that uses them to select a controller and the alleles are saved in preparation for a new window of training data.

Since new feedback data is available from the sensors every five seconds the system was designed to run on a new window of data every five seconds. Normally a genetic algorithm or any estimating algorithm would receive a stopping criteria based on error. In this case the stopping criteria is based on number of generations due to the time constraints and the need to analyze new data every five seconds. To meet this requirement the system was limited to 10 generations for each window and parallel computing techniques were used to minimize computation time.

One issue to consider when estimating parameters is the occurrence of local minimums. This GA approach to parameter estimation was chosen for its ability to avoid local minimums compared to some traditional search methods. Search methods can easily find local minimums and have no means of correcting to find a best solution. An example of a local minimum for this application is the combination of 6 parameters being estimated could produce a relatively low error with a certain set of alleles but that set is not necessarily the best or correct. This is noticed more prominently when the randomly chosen initial guesses are far away from the appropriate parameters. When this occurs

the GA struggles to find the best set of parameters. Increasing the mutation rate can help combat this but that solution can also introduce unwanted oscillations in the output or cause the algorithm to continue to fluctuate dramatically when a good solution is found.

To combat this phenomena a structured system of predefined guesses are systematically tested every time new data is given. The disturbance input parameters remain unchanged but each generation is tested against one other nominal plant as defined in Table 1 and Table 2. This is done systematically in order to offer the genetic algorithm a better starting point if the mutation of one or multiple parameters begins to run off or if the system has stagnated at a local minimum. The decision to follow one of the predefined options is based on the same fitness calculation as the other sets of parameters.

After each window of feedback data has gone through the genetic algorithm the best set of parameters is used to select a nominal plant, which will be used to select a controller, based on that nominal plant. The gain and time constant of the FiO_2 transfer function determine the nominal plant and therefore the controller that is selected.

The parameter estimation process is represented in Fig. 6. This demonstrates the continuously updating nature of the parameter-estimating model. When the system is started an array of 6 parameter sets are randomly assigned within the appropriate range. Upon each iteration a default set of alleles is also provided for analysis along with the random. The fitness of each individual is determined in comparison to the feedback data. The sets are chosen to be mated with the roulette function then mated with the crossover function, but found in Appendix D. Randomly selected individual parameters are mutated before the process is complete and the new groups of alleles are passed back for

the next generation and the best set is saved. The window of feedback information is updated every 5 seconds so the process is started over again with the goal of finding the best set of parameters to represent the model.

When the automatic control of FiO_2 is turned on the genetic algorithm estimator works the same way. The controller chooses the FiO_2 set point but the estimator receives all the feedback information and selects the controller to be used as described above. The model used to obtain the closed loop simulation results operates as seen in Fig. 7. This is also the same manner that the control system will work in a clinical setting except that the feedback information will be from an actual patient instead of being simulated with a model using known parameters.

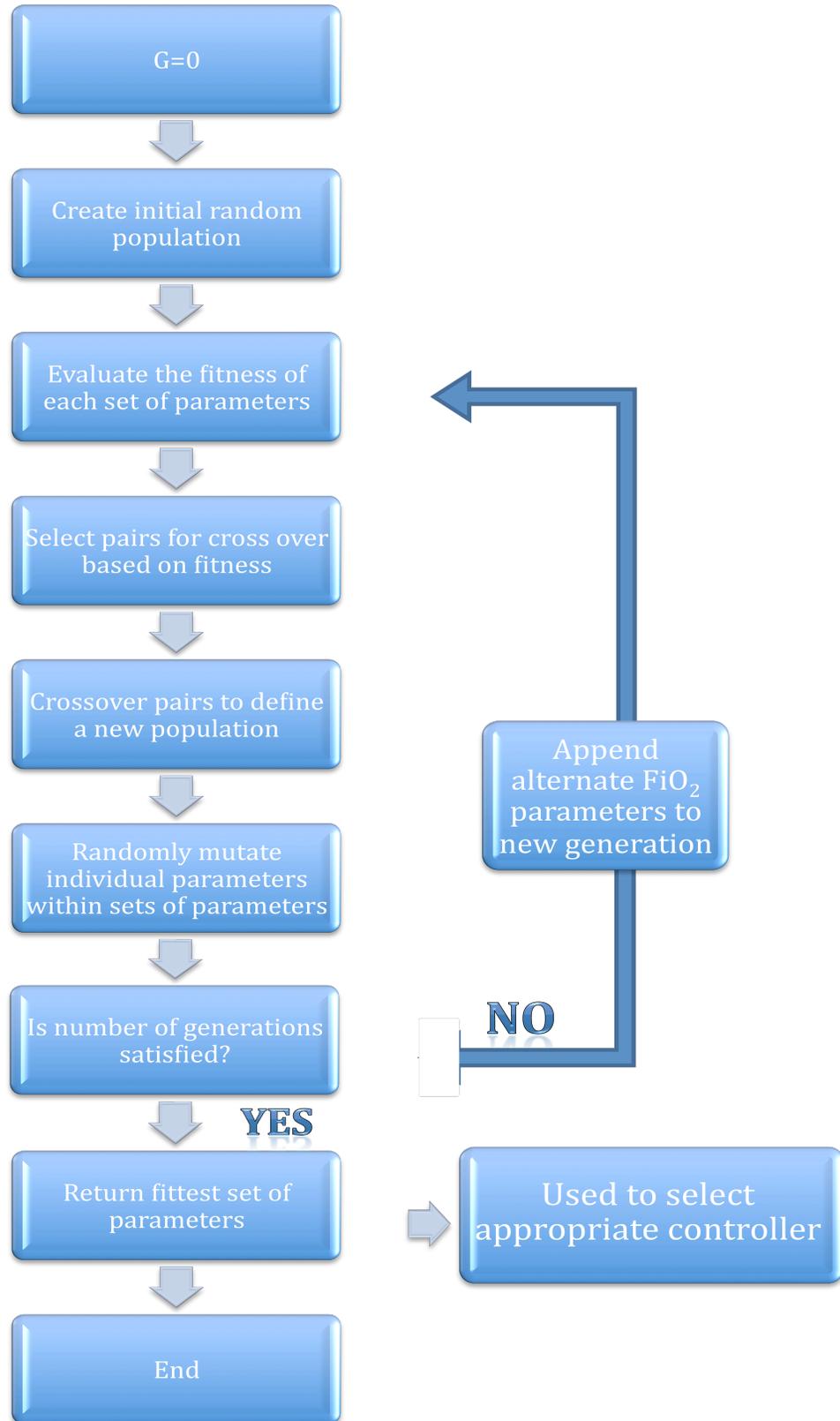


Fig. 6. Flowchart of the parameter estimating genetic algorithm model

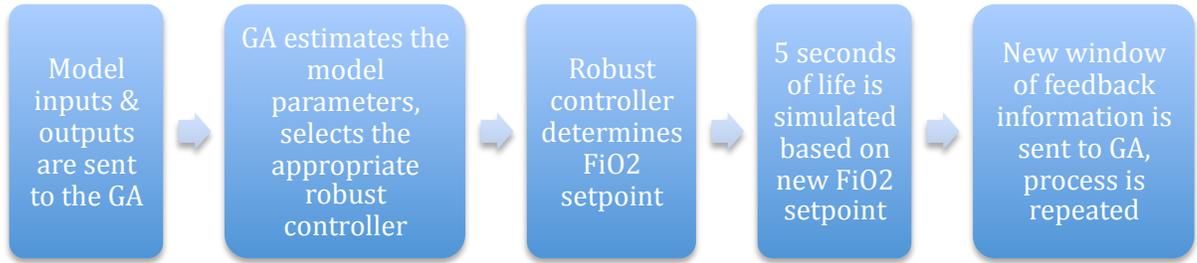


Fig. 7. Parameter estimation simulations with closed loop FiO₂ control

2.4 Non-linear model

Constitutive models have been developed to describe the FiO₂ – SpO₂ relationship, which are based on the O₂ dissociation curve. One such model is able to more accurately model a patient’s response [24]. The oxygen dissociation equation, which the non-linear model is based on, is described as

$$SpO_2 = \frac{1}{23400(p_a^3 + 150p_a)^{-1} + 1} (100\%) \quad (3)$$

where P_a is the partial pressure of oxygen in the artery. This equation was adapted to model preterm infants and appropriate constants were determined in previous work and can be seen in Fig. 8[29]. The derivative is computed at regular intervals to find the gradient describing the absorption of oxygen into the bloodstream. As the hemoglobin in the blood is closer to fully saturated with O₂ molecules its affinity for Oxygen is reduced. Therefore the higher the saturation the harder it is to absorb more oxygen and the less saturated the easier it is for the blood to uptake more Oxygen.

While more accurate, this model includes non-linear equations that significantly inhibit the computational speed of simulations, as well as hinders the use of established

control design techniques. It also limits the ability to validate the genetic algorithm selection process since the actual model parameters do not match the estimations process. For these reasons the non-linear model was used only as a secondary testing method for acquiring simulated closed loop results. To do this the non-linear model was simply substituted in place of the FiO_2 transfer function in the three transfer function model. In this way the disturbance inputs are still used but the $FiO_2 - SpO_2$ relationship is more accurate.

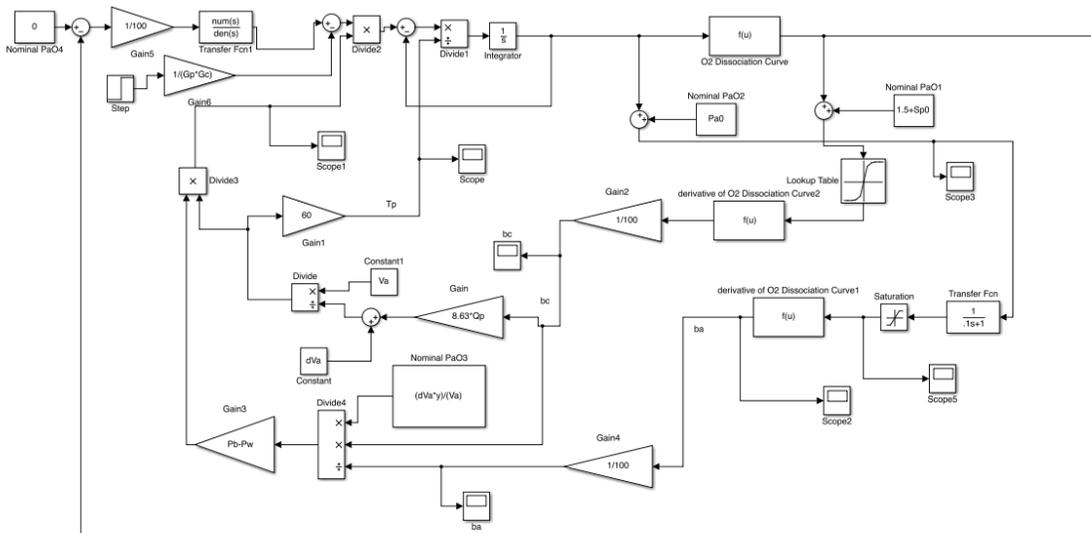


Fig. 8. Block diagram of non-linear model of $FiO_2 - SpO_2$ relationship

Chapter 3: Control design

3.1 Performance Specifications

The model was broken into six ranges of parametric uncertainty as described with the intent of designing a controller for each range. Each controller is designed to adjust the FiO_2 in a manner that emulates the best practices of an attending nurse. The protocols used in the NICU that have been clinically proven to reduce the rate of ROP in low birth weight infants are to be used as guidelines for defining the controller operation using μ -synthesis techniques to find H-infinity controllers. H-infinity control was chosen for the ease of implementation in terms of the body of theoretical work that allows for clearly defined controller performance as well as robust stability and performance analysis. The synthesis methods used here and described in detail allow the strict control of key aspects of the performance which are vital to the patient's health. Other considered methods such as PID control pose unique problems such as gain tuning as patient physiology changes and possible calculation errors when sudden desaturations occur.

The protocol for manual FiO_2 management is summarized as [7]:

- Oxygen is a drug: FiO_2 should be minimized at all times as saturation levels above 95% are potentially dangerous
- FiO_2 should be supplied to maintain a minimum saturation of 85%
- FiO_2 will not be changed frequently “up and down and up again” to maintain acceptable levels
- Weaning should be done gradually if the patient remains on the “high” side but can be done quickly to avoid hypoxia
- Do not keep an increased FiO_2 unless necessary

- Small incremental adjustments of 2-5% are best, do not vary FiO_2 in large steps
- Alarms will not be turned off at any time. Settings must not be changed due to frequent alarms or adjustment in FiO_2 .
- FiO_2 requirement of the infant should be documented clearly

This protocol was then translated into frequency domain performance specifications that can be used to define how the controllers operate. The controller will be stabilizing the closed loop feedback control system while achieving the frequency response specifications on the control effort and error responses. As part of the μ -synthesis controller design procedure the performance weight $w_u(s)$ is defined as

$$w_u(s) = \frac{1}{u_{\max}} \quad (4)$$

where u_{\max} is the maximum control effort. Assuming a maximum of 100% FiO_2 and a nominal value of 21% oxygen in the air the value of u_{\max} is 79. The use of this performance weight also helps minimize the control effort, which coincides with the protocol of minimizing FiO_2 and “weaning”. In some cases a hospital may dictate a maximum FiO_2 that is below 100%. In those cases it would be very simple to go back and alter the value of u_{\max} to synthesize a controller that meets that requirement.

According to the literature it is important to facilitate quick recovery from hypoxic events, or desaturation of the SpO_2 , but also important to minimize adjustments of the FiO_2 . The second performance specification helps define how the controller will respond to higher frequency events like fast desaturation, lower frequencies close to

steady state, and the crossover bandwidth which differentiates the two. This performance weight, called $w_p(s)$, is defined as

$$w_p(s) = \frac{\frac{s}{M} + \omega_b}{s + \omega_b A} \quad (5)$$

where ω_b is the bandwidth, M is the high frequency error allowable, and A is the low frequency error allowable. Using this weight defines the maximum steady state error that is acceptable, the maximum high frequency error, and the crossover bandwidth. Since the acceptable range for SpO₂ is 85%-93% if we make the controller set point 93%, the maximum steady state error is 8% SpO₂. If we want stricter control at low frequencies we can reduce the value of A to 5% or 3%. The crossover bandwidth can be set to define what frequencies will affect the control signal and what frequencies will be rejected as disturbances. This allows us to meet more of the FiO₂ management requirements by limiting the FiO₂ response to low frequency oscillations or sensor noise. This means less changing the FiO₂ “up and down and up again” frequently and not keeping an increased FiO₂ setting unless necessary.

The third performance weight used is the key to limiting the rate at which the FiO₂ adjustments are made. Since it is important to adjust the FiO₂ slowly and in small increments as opposed to large steps the rate limiting performance weight is defined as

$$w_i(s) = \frac{1}{\dot{i}_{\max}} \frac{s}{\tau s + 1} \quad (6)$$

where \dot{i}_{\max} is the maximum rate of change of the control effort and τ is the time constant of the approximate derivative and should be sufficiently small to achieve an acceptable result. The approximate derivative must be used in this case to ensure that the

performance weight is invertible when being used to synthesize the controller. For the controller synthesis, τ was given a value of 1.0×10^{-15} . This performance weight helps limit the rate at which the FiO_2 is increased or decreased. In other words it is a maximum for the first derivative of the FiO_2 . This meets the protocol of varying the FiO_2 in small incremental adjustments instead of large steps. Since our hardware allows the FiO_2 to be controlled continuously, or in small time steps such as 5 seconds, and more precisely than a nurse, the FiO_2 can be controlled in a more optimal way in terms of the infant's health. A nurse may only be able to determine a resolution of 1-2% FiO_2 and might only be able to adjust the FiO_2 every few minutes. This performance specification helps the controller match the same result but with slow continuous adjustments.

For controller synthesis the performance weights and uncertainty models are applied to the three transfer function model with the RR and HR aspects re-arranged and labeled as disturbances. This can be seen in Fig. 9 along with the controller K which is to be determined. In this figure the W blocks for HR, RR and FIO_2 represent the uncertainty model that is multiplied by the Δ block which is some transfer function with $H_\infty < 1$ and is stable and proper.

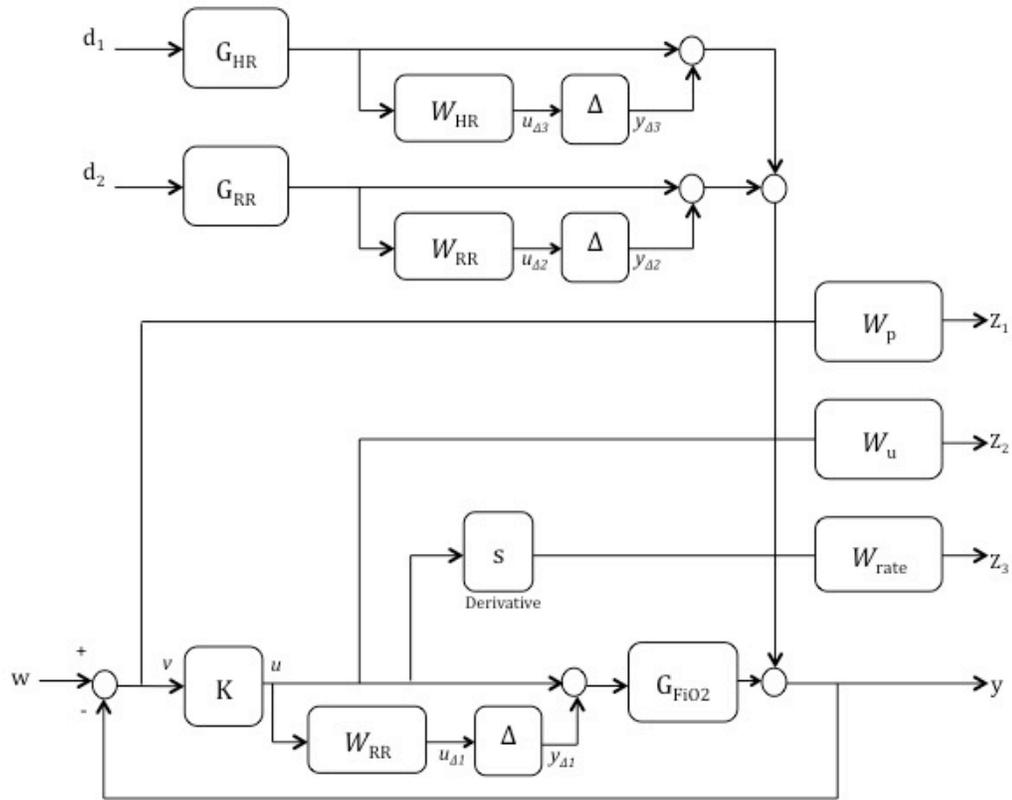


Fig. 9 Block Diagram of model, performance weights, and uncertainty model

3.2 H-infinity synthesis

The controller K is to be synthesized such that the closed loop system is stable and the H-infinity norm between the inputs and outputs of the matrix in Fig. 9 are less than one. If this condition is met then the resulting controller is said to have robust performance, which means that for the defined range of uncertain model parameters the close loop control meets the desired performance specification. The μ -synthesis process to find the actual controllers was done using the `hinfyn.m` command in MATLAB®. This command uses a mixed sensitivity approach, which minimizes the cost function

containing the three performance weights from above and this cost function must be less than one to meet performance specifications as seen in equation (7)[30].

$$\left\| \begin{array}{c} w_p(s)S \\ w_u(s)KS \\ w_{ii}(s)T \end{array} \right\|_{\infty} < 1 \quad (7)$$

where S is the sensitivity transfer function of the closed loop system, and T is the complementary sensitivity function.

The generalized plant transfer function matrix, P , that represents the system in Fig. 9 and is used for the controller synthesis process is shown in Equation (8). Once this synthesis process was completed for each of the 6 controllers verification of their performance is completed to ensure the performance specifications were met.

$$\begin{aligned}
P &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & W_{FiO_2} \\ 0 & 0 & 0 & 0 & W_{RR}G_{RR} & 0 \\ 0 & 0 & 0 & W_{HR}G_{HR} & 0 & 0 \\ -w_p G_{FiO_2} & -w_p & -w_p & -w_p G_{HR} & -w_p G_{RR} & -w_p G_{FiO_2} \\ 0 & 0 & 0 & 0 & 0 & w_u \\ 0 & 0 & 0 & 0 & 0 & w_{\dot{u}} \frac{1}{s} \\ -G_{FiO_2} & -1 & -1 & -G_{HR} & -G_{RR} & -G_{FiO_2} \end{bmatrix} \\
P_{11} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & W_{RR}G_{RR} \\ 0 & 0 & 0 & W_{HR}G_{HR} & 0 \\ -w_p G_{FiO_2} & -w_p & -w_p & -w_p G_{HR} & -w_p G_{FiO_2} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\
P_{12} &= \begin{bmatrix} W_{FiO_2} \\ 0 \\ 0 \\ -w_p G_{FiO_2} \\ w_u \\ w_{\dot{u}} \frac{1}{s} \end{bmatrix} \\
P_{21} &= \begin{bmatrix} -G_{FiO_2} & -1 & -1 & -G_{HR} & -G_{RR} \end{bmatrix} \\
P_{22} &= \begin{bmatrix} -G_{FiO_2} \end{bmatrix}
\end{aligned} \tag{8}$$

3.3 Performance Analysis

Verification was done to ensure that each controller, K , found by the μ -synthesis technique meets robust stability requirements and the performance specifications. To ensure the controllers are nominally stable with no plant uncertainty and there must be no right-half plane poles for the nominal closed loop plant. One performance check was to ensure the value of Equation (7) was less than one. Another method includes analyzing the N matrix. This is found through the lower fractional transformation of the P matrix described in Equation (8) and the controller K . The inputs and outputs of the N matrix follow the labeling scheme of Fig. 9. and are shown as

$$\begin{Bmatrix} y_{\Delta 1} \\ y_{\Delta 2} \\ y_{\Delta 3} \\ Z_1 \\ Z_2 \\ Z_3 \end{Bmatrix} = \begin{bmatrix} N_{11} & N_{21} \\ N_{12} & N_{22} \end{bmatrix} \begin{Bmatrix} u_{\Delta 1} \\ u_{\Delta 2} \\ u_{\Delta 3} \\ d_1 \\ d_2 \\ w \end{Bmatrix} \quad (9)$$

Nominal performance checks that the nominal plant is controlled according to the performance specifications. This is done by ensuring that the H-infinity norm of the lower partition of the N matrix is less than one,

$$\text{Nominal performance: } \|N_{22}\|_{\infty} < 1 \quad (10)$$

Robust stability and robust performance ensures that the closed loop system is stable and meets the given performance requirements over the given range of uncertain parameters. The necessary condition for robust stability is given as,

$$\text{Robust Stability: } \|N_{11}\|_{\infty} < 1 \quad (11)$$

The final test is robust performance, which ensures performance specifications are met for all uncertain plants. This is confirmed if the N matrix has a maximum singular value less than one.

$$\text{Robust performance: } \|N\|_{\infty} < 1 \quad (12)$$

The performance criteria used to define the performance weights in some cases was different than the performance weights used to calculate the performance checks.

This was done in instances when the optimization process for the control synthesis did not find a controller that met the original performance specifications. Modified performance weights were applied in order to ultimately find the desired performance. In all cases the controllers were able to meet the original criteria required but fine-tuning the synthesis process was required.

Chapter 4: Results

4.1 Controller Synthesis results

The synthesis of the six controllers was completed as described in the previous chapter. To verify that each controller, K , met the performance criteria that were initially specified, Equation (7) was checked and confirmed. The criteria outlined in Equations (10) - (12) were checked for all six instances but did not prove robust stability and robust performance for all disturbance inputs. When the disturbance input uncertainty models W_{HR} and W_{RR} are reduced to zero the system does meet the robust performance criteria. The performance criteria results for the controllers without considering the disturbance uncertainty are show in Table 3. A full description of the controller transfer functions and an example bode magnitude plot can be found in Appendix B.

Table 3 Performance and Stability verification values

Plant #	Nominal Stability (# RHP)	Nominal Performance	Robust Stability	Robust Performance
1	0	0.26	0.49	0.993
2	0	0.24	0.58	0.995
3	0	0.23	0.32	0.990
4	0	0.30	0.38	0.997
5	0	0.28	0.36	0.996
6	0	0.27	0.36	0.996

It is also useful to see the actual controller response to understand the benefit of each performance specification. To ensure that the controllers were operating in a desired manner and that the rate limiting performance specification was working as desired a simple step command was given to the system. In Fig. 10 the control signal

response to a step input is shown when a controller is synthesized without using the rate limiting performance specification. It should be noted that the control effort instantly jumps up to 12% FiO₂ then quickly drops to a small steady state value. This means a large fluctuation in FiO₂ just to get a quick recovery that is most likely unnecessary, and possibly harmful to the baby's health.

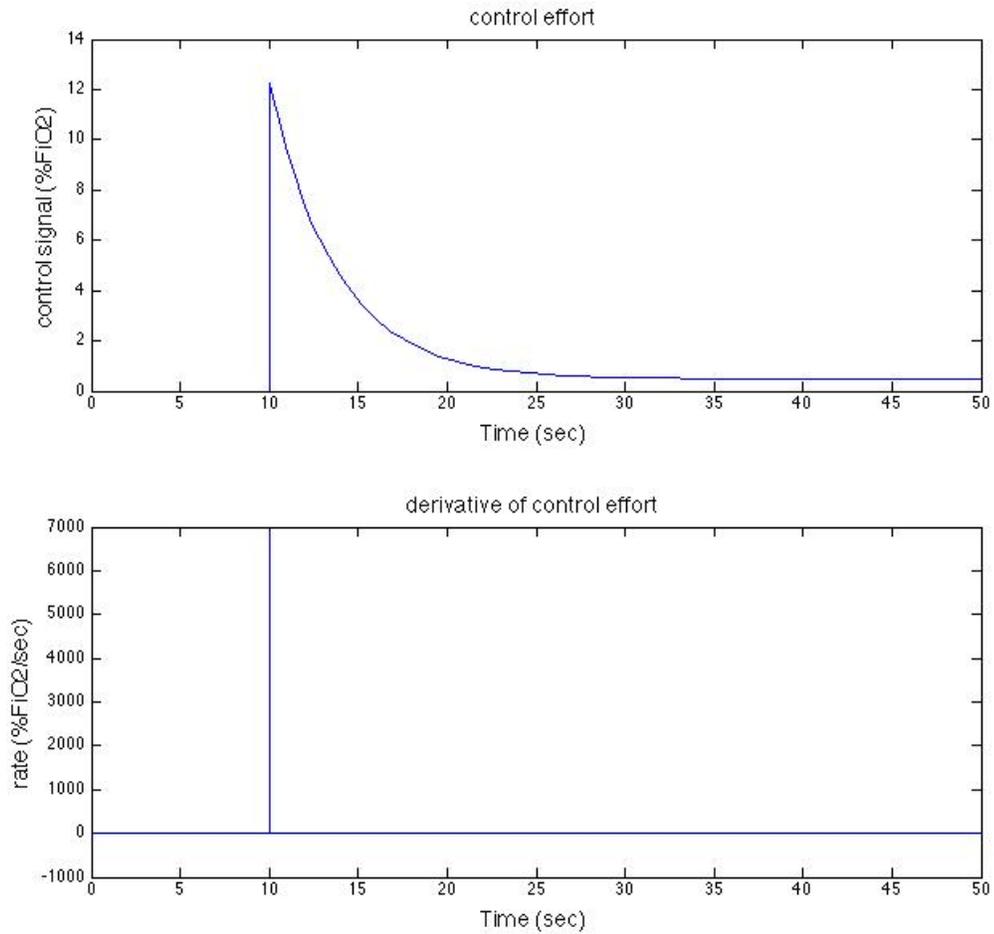


Fig. 10. Closed loop step response of controller without rate limiting performance specification

Compare the results in Fig. 10. to the response of a controller that was synthesized with the addition of the rate limiting performance specification, which can be seen in Fig.

11. This response varies greatly in terms of the maximum control effort achieved, as well as the rate in which the FiO_2 was varied. In both cases the same steady state error and same steady state control effort was achieved. In the second case the system took slightly longer to achieve steady state. That is acceptable according to our goals and from the perspective of the patient's health, potentially very beneficial.

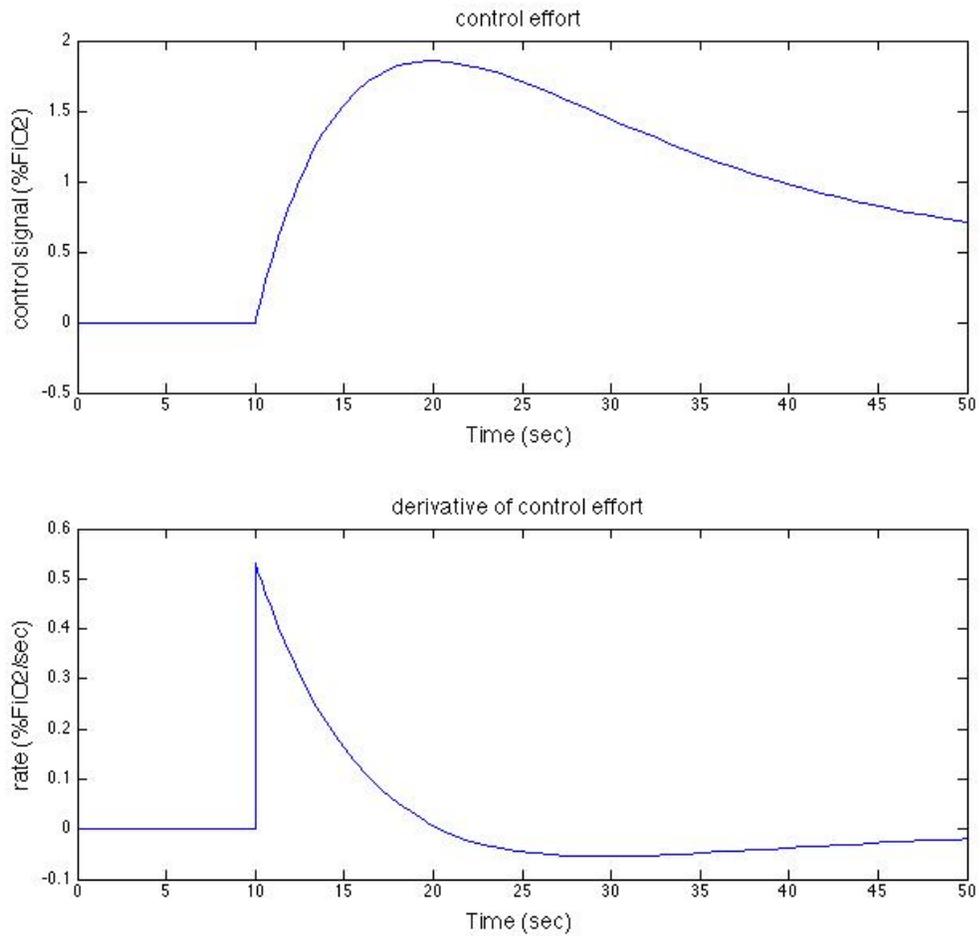


Fig. 11. Response of a controller synthesized using the rate limiting specification

4.2 Open loop parameter estimation

This approach for controlling the premature infant's SpO_2 is based on the ability to accurately estimate the model parameters and select the appropriate robust controller. It does this by matching the best set of parameters to one of the six nominal plants as described previously. For this approach to work the algorithm must be able to pick the best controller reliably and the controller must function properly.

The effectiveness of the genetic algorithm in selecting the appropriate controller was determined using a series of tests. Since it is impossible to verify the actual model parameters of clinically collected data, a way of creating simulated SpO_2 to validate the GA was developed. Hour long strings of clinically collected FiO_2 , HR, and RR data were selected as the inputs to the model. Model parameters were then chosen for the three transfer functions and the input data was used to obtain a simulated SpO_2 response. This data can now be used to verify the efficacy of the genetic algorithm because the inputs are realistic and the model parameters are known.

The input data and SpO_2 are given to the genetic algorithm using its updating window approach to parameter estimation. As the window moves up every 5 seconds a new set of information is provided to the genetic algorithm. It runs 10 generations and provides a set of 6 model parameters, gain and time constant of the transfer functions that describe each input, that correspond to the lowest error. Since the FiO_2 gain and time constant are used to select which controller is appropriate, it is important that their parameter estimation is accurate. This selection is given in the form of the nominal plant number, 1-6, where each plant is described previously in Table 1 and Table 2.

4.2.1 Constant parameter estimation

Simulations were arranged with parameters falling in the range of each of the 6 controllers to test the ability to select each one accurately. The three transfer function model was used for this simulation so that the known model parameters could be compared to the estimated ones. The data was prepared with a time step of 0.1 seconds to match the time step of the GA and ensure stability. The parameter estimation for a case where controller number one is the correct solution is shown in Fig. 12. The red horizontal lines indicate the correct parameter that is to be estimated and the blue represents the best guess at each iteration. This simulation was done for 200 iterations of the genetic algorithm, which represents a little over 16 minutes of life for the baby.

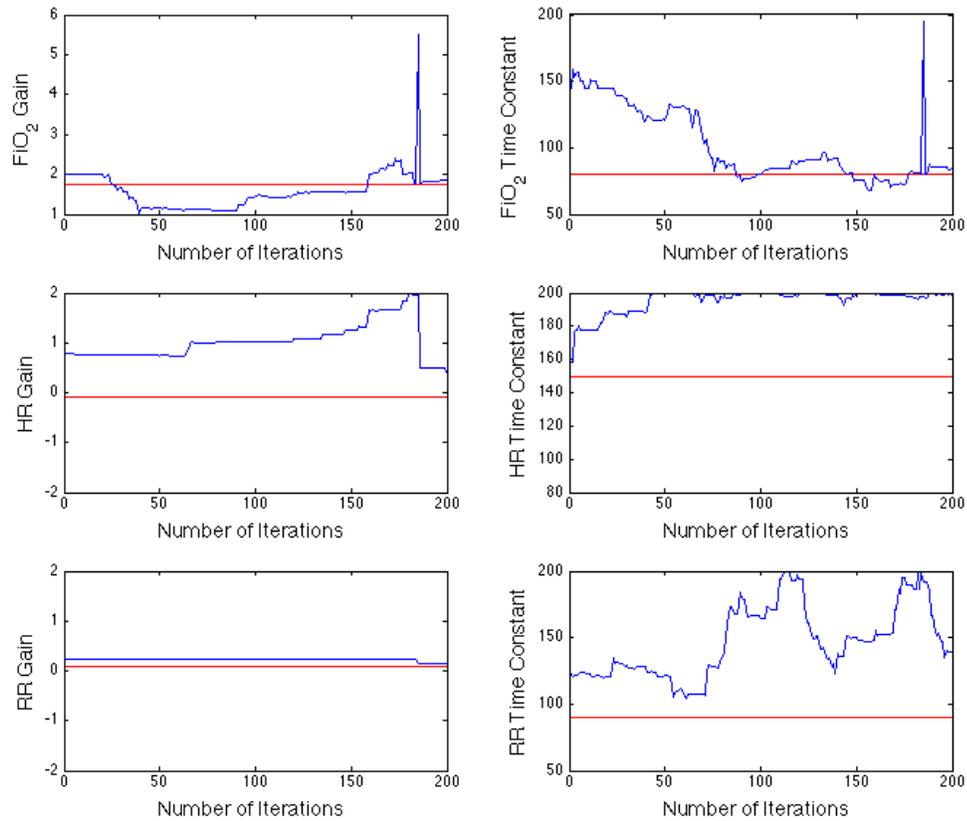


Fig. 12. Controller 1 parameter estimation simulation

The result in Fig. 12 shows that the FiO_2 transfer function gain is estimated to an acceptable level of accuracy by the first iterations of the algorithm and maintains a close approximation the entirety of the simulation. In terms of selecting the best controller the gain portion of the algorithm is only responsible for deciding if the gain is above or below a value of 3. In most cases the gain estimation is accurately estimated within 20 iterations. The time constant was randomly assigned an initial guess that was substantially further away from the correct parameter but the result shows how it continues to improve over the next 10 minutes of simulated life and eventually finds the

correct solution by the 70th iteration. The error calculated for each moving window of data is shown in Fig. 13. which illustrates the larger error associated with the poor parameter estimation near the end of the simulation in Fig. 12.

A representation of which controller the algorithm selected during this simulation is show in Fig. 14. As can be seen, it ended up on the correct controller except for one time step when it estimated a different plant but then returned to the right choice on the next iteration. This happens occasionally and is a built in aspect of the genetic algorithm. Mutations such as this allow exploration of other options and provide opportunities to jump out of local minimums. The random selection of a different controller that could alter the optimal control of FiO_2 is easily remedied by a requirement for a sustained choice for a defined amount of time or averaging which would eliminate outliers while still allowing the GA the necessary freedom to make those mutations.

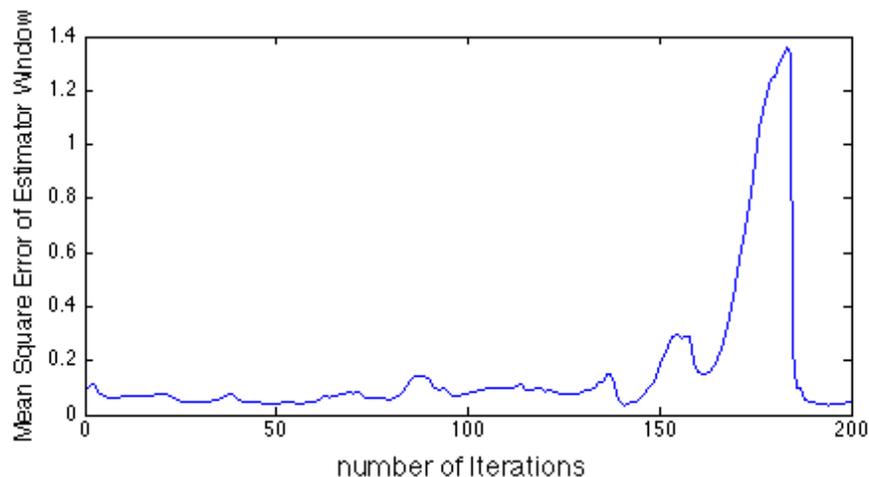


Fig. 13. Error used to assign fitness of parameter sets during simulations

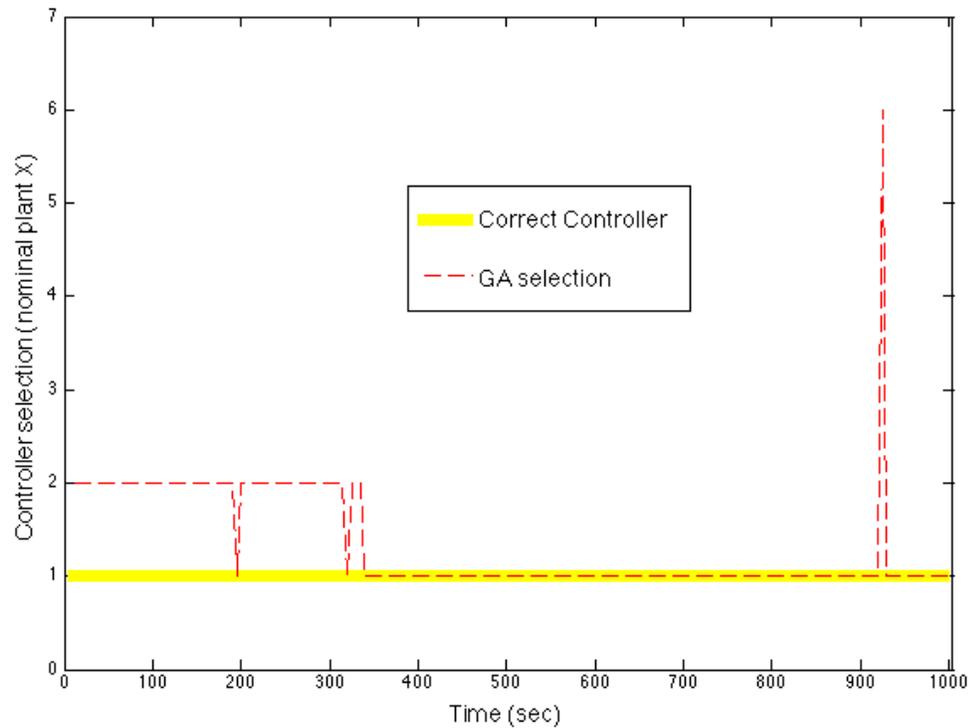


Fig. 14. Yellow line represents the correct choice, red is simulation result

The other 6 controllers exhibit very similar results when the correct model parameters are close to the nominal plant parameters. Occasionally when the correct parameters are between two plants the best guess will fluctuate between those two controllers. Since the ranges that the error models and controllers were designed with overlap, this oscillation or indecision is not a problem in terms of robust stability. If the correct model parameters lie in the overlap range of the robust controllers then both choices will operate with robust performance. The result when the correct time constant is between controllers 2 and 3 is shown in Fig. 15.

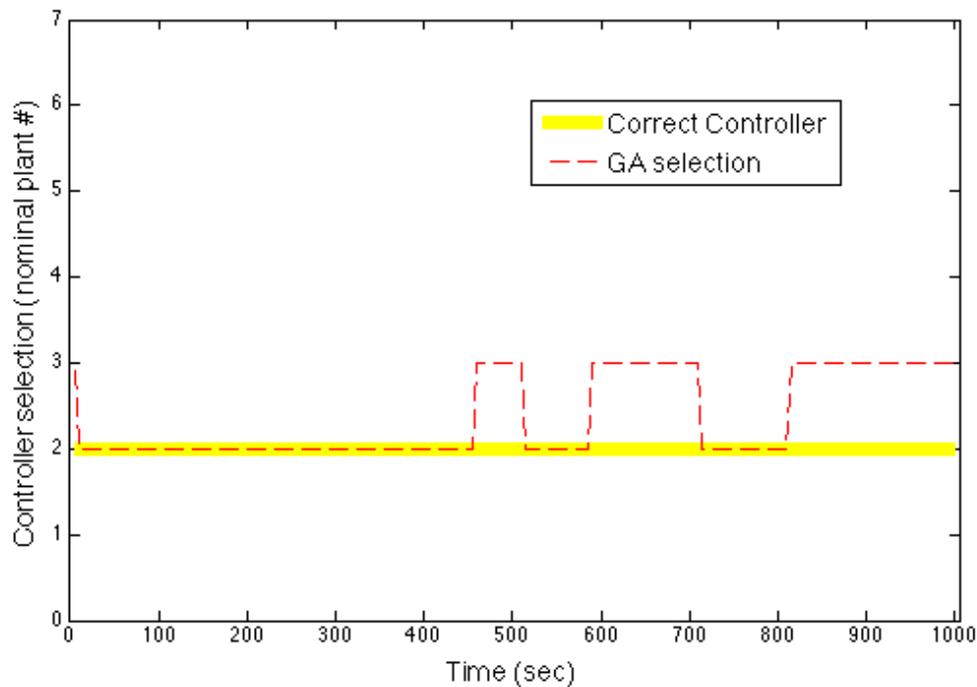


Fig. 15. Result when the correct parameters are between controllers

4.2.2 Changing parameter estimation

The genetic algorithm must also be able to adapt to changing parameters, not just select one set reliably. To test its ability to adapt to changing parameters and simulate its tracking of physiological changes in the patient another test was devised. An hour-long set of inputs was fed into the three transfer function model as before but at specified intervals the time constant of the FiO_2 transfer function was changed. The window of inputs and response then represented changing parameters that the GA must estimate.

The first test was to ensure it could estimate increasing time constants. The FiO_2 gain remained constant while the time constant increased from 80 to 145 and then 185. This was done over 50 minutes and the time constant increased every 1000 seconds. The results of this test can be seen in Fig. 16 and Fig. 17.

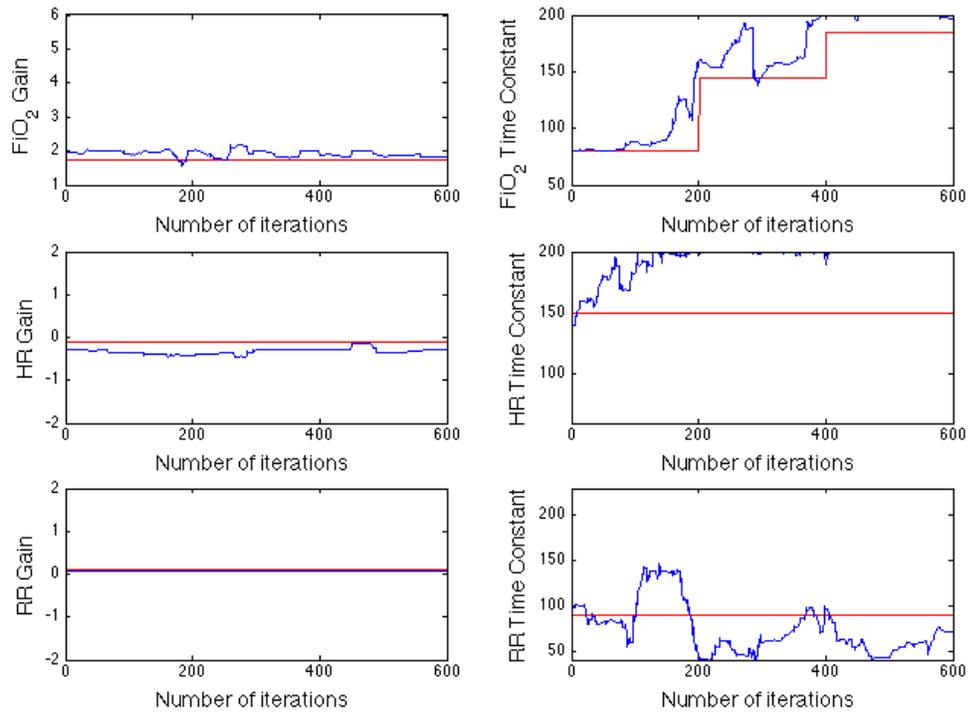


Fig. 16. Estimating varying time constants test

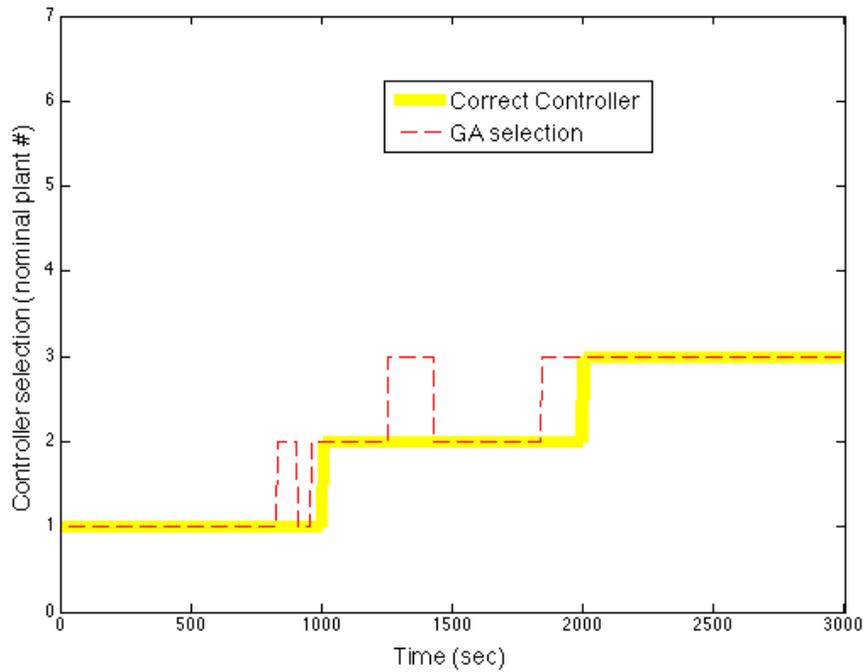


Fig. 17. Controller selection for varying time constant test

The gain selection in this result is very accurate as can be observed in the top left plot of Fig. 16. The time constant estimation in the top right doesn't have perfect accuracy but it does track the changing parameters. After it drifts away from the best solution and occasionally selects the wrong nominal plant, it is able to correct itself and return to a better solution. The time constant selection appears to anticipate the change in parameters but this is a result of the window of past data being used to estimate.

This simulation, Fig. 16 -Fig. 17, for showing the ability to adapt to changing model parameters is a little unrealistic compared to what you might see in the physiology of a real baby. Such drastic increases in the time constant of FiO_2 response would not occur two times in one hour. This was done merely to illustrate the functionality of the system in as simple a way as possible. If it can estimate large parameter changes quickly it will be able to detect small gradual changes that much easier.

Unpredictable changes in patient physiology that cause the time constant of the response to shorten, although rare, can happen. To show the ability to switch between the remaining three controllers and also the ability to estimate time constants that are decreasing another simulation was developed. The parameter estimations for the test that changes from plant 6 to 5 to 4 can be seen in Fig. 18 and the controller selection can be seen in Fig. 19.

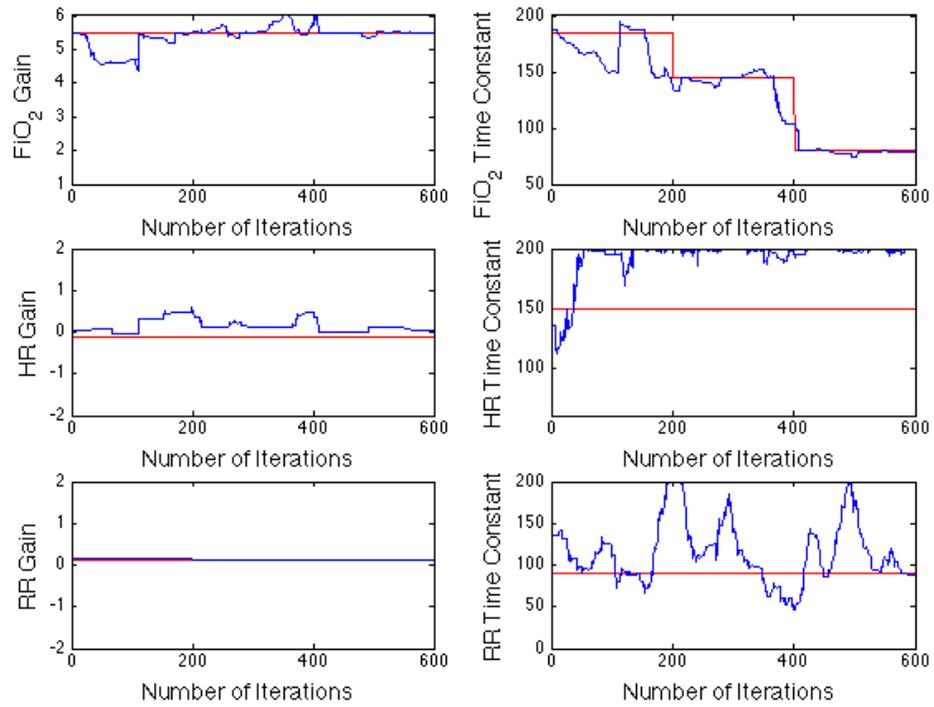


Fig. 18. Estimating varying time constants test 2

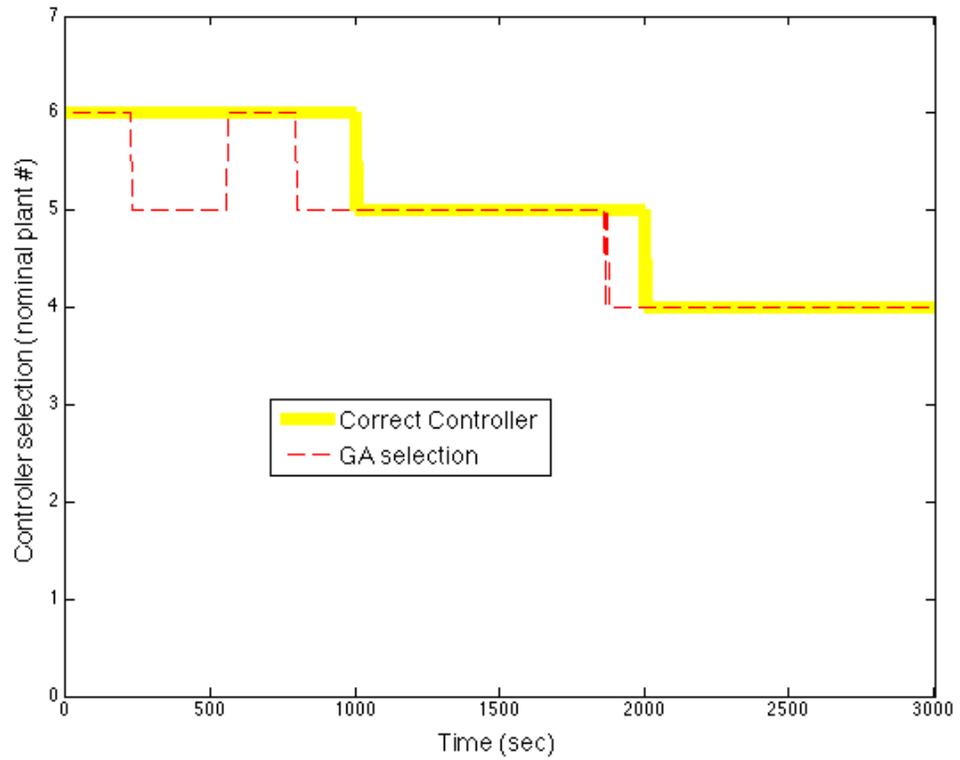


Fig. 19. Controller selection, varying parameters test 2

As the patient grows and gains weight the volume of red blood cells increases, muscle mass increases, and other physical characteristics change. This could lead to changes in the system gain. A test to illustrate the model's ability to adapt to changing system gains is seen in Fig. 20 and Fig. 21. The estimator is able to track the changing model parameters and select the appropriate controllers.

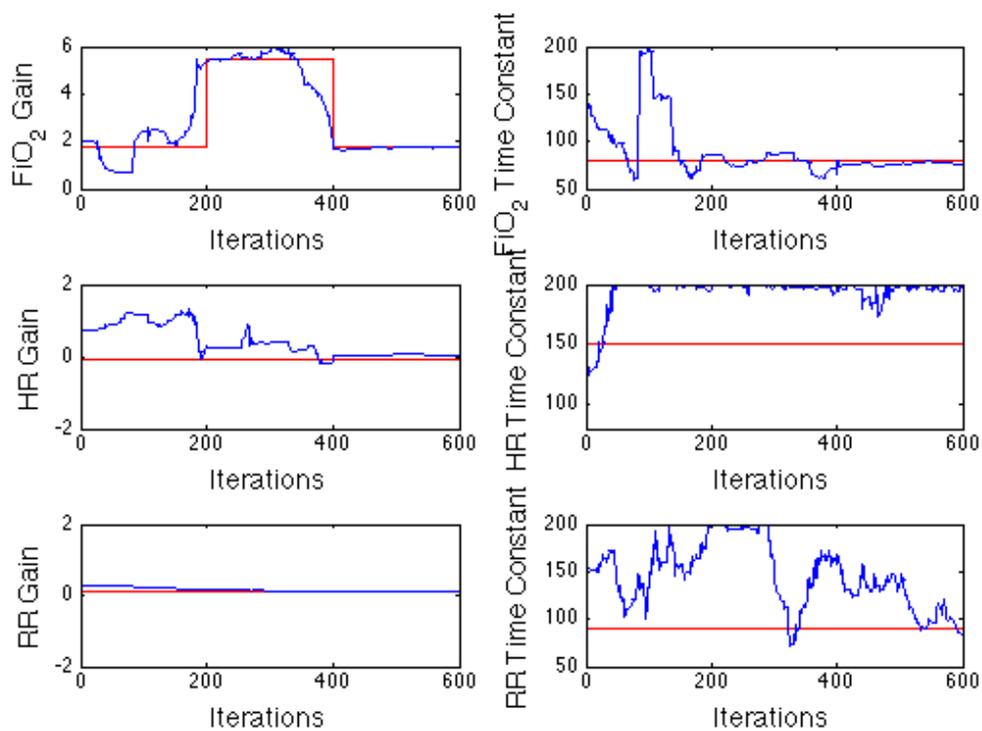


Fig. 20. Changing gain parameter estimation

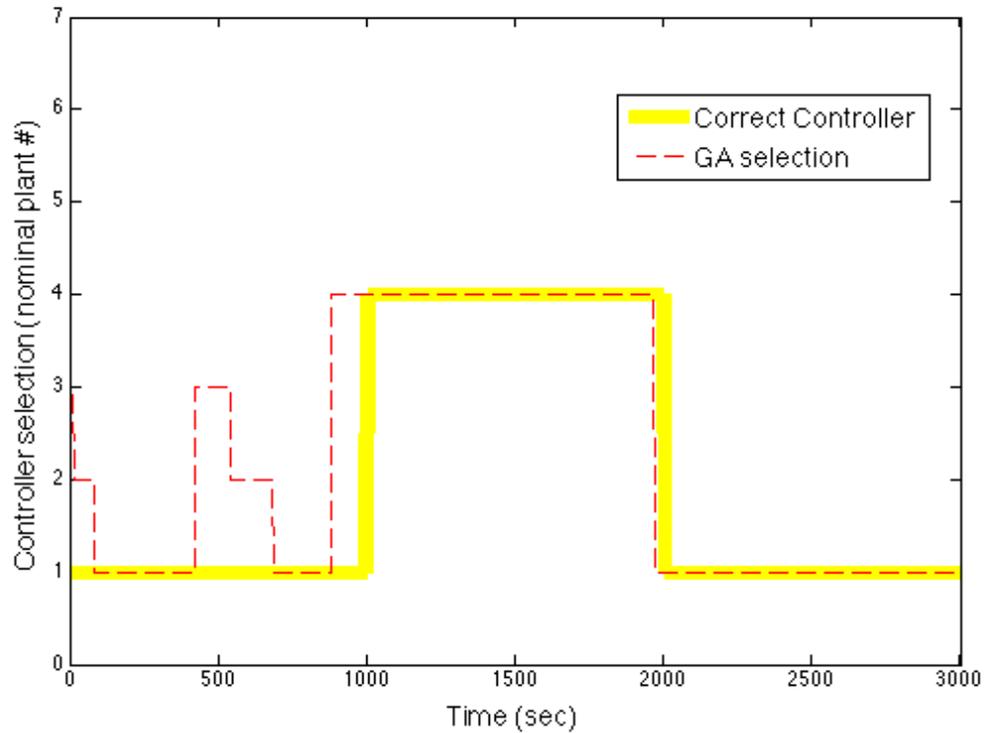


Fig. 21 Changing gain controller selection

4.2.3 Parameter estimation with real data

Other means of verifying the viability of the genetic algorithm were employed to ensure that it has the ability to operate in a clinical environment with real patients. To do this the same algorithms were applied to sets of real inputs as before. However, the SpO₂ data was not simulated this time. Since the genetic algorithm is running on real data these simulations will give insight to real time operation but verification of parameter selection is impossible.

Hour long sections of data with a high number of FiO₂ changes were selected to ensure adequate excitation of the transfer functions. The raw data was sent to the genetic algorithm with a window of 5 minutes, which proved to give the most consistent results. An abbreviated example of these results is shown in Fig. 22 so that more detail can be

seen. There is no red line to indicate the correct parameters since they are unknown. It was noted that the magnitude of the mean square error in this case was on the order of 10 to 50 times higher than the results above where the SpO_2 was simulated. This is to be expected since the simulated SpO_2 was created from the model that was used to do the estimating. The model will never match the actual patient response perfectly since it is only simulating a very simplified representation of the patient. The controllers selected during this test are shown in Fig. 23. and the input data is shown in Fig. 24.

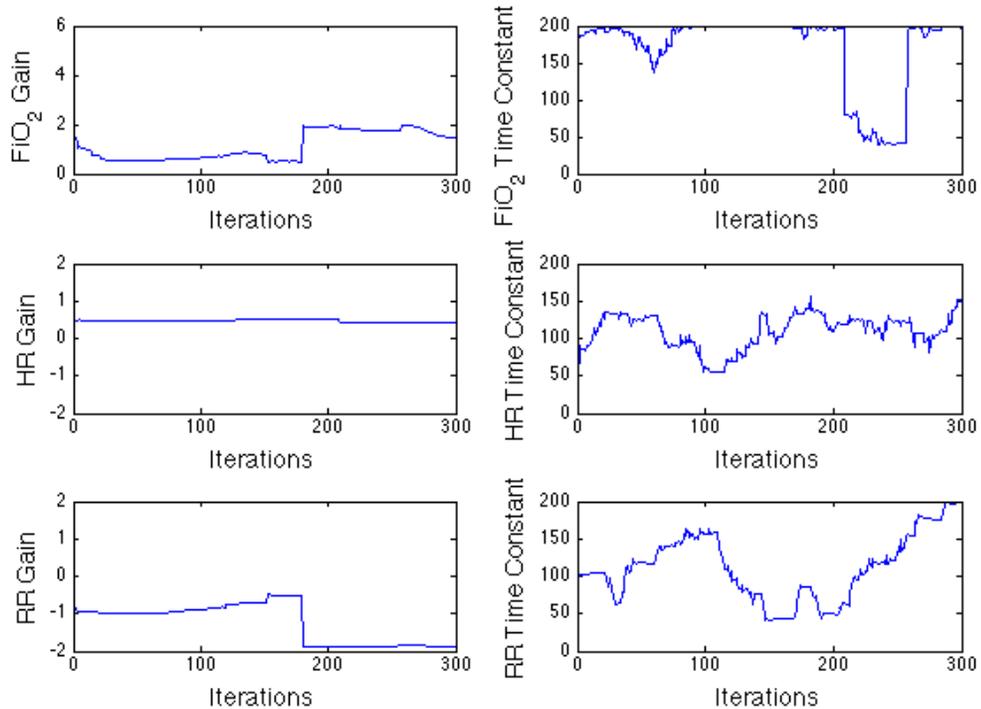


Fig. 22. Raw data simulation with unknown model parameters

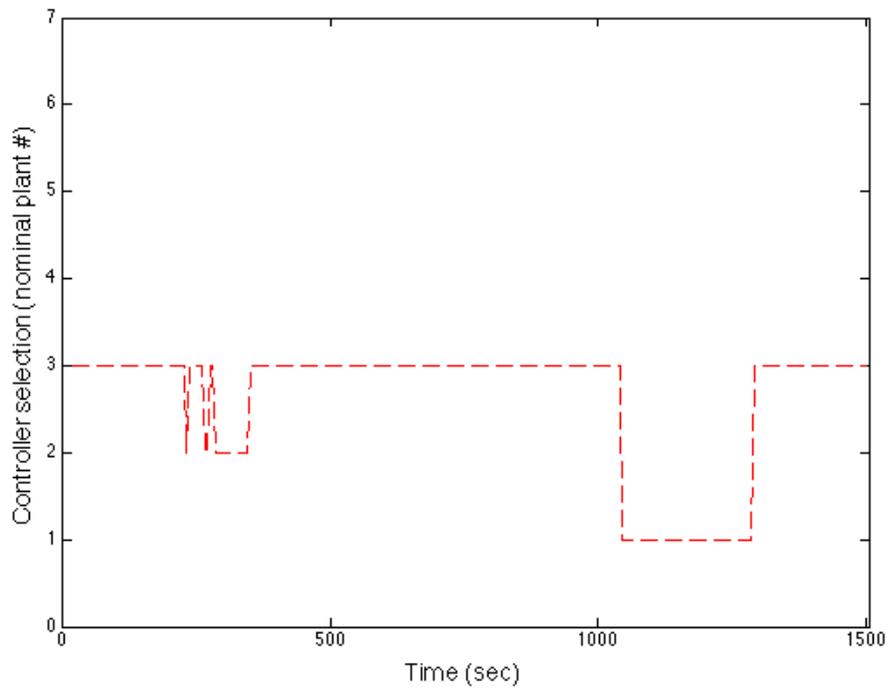


Fig. 23. Raw data simulation controller selection

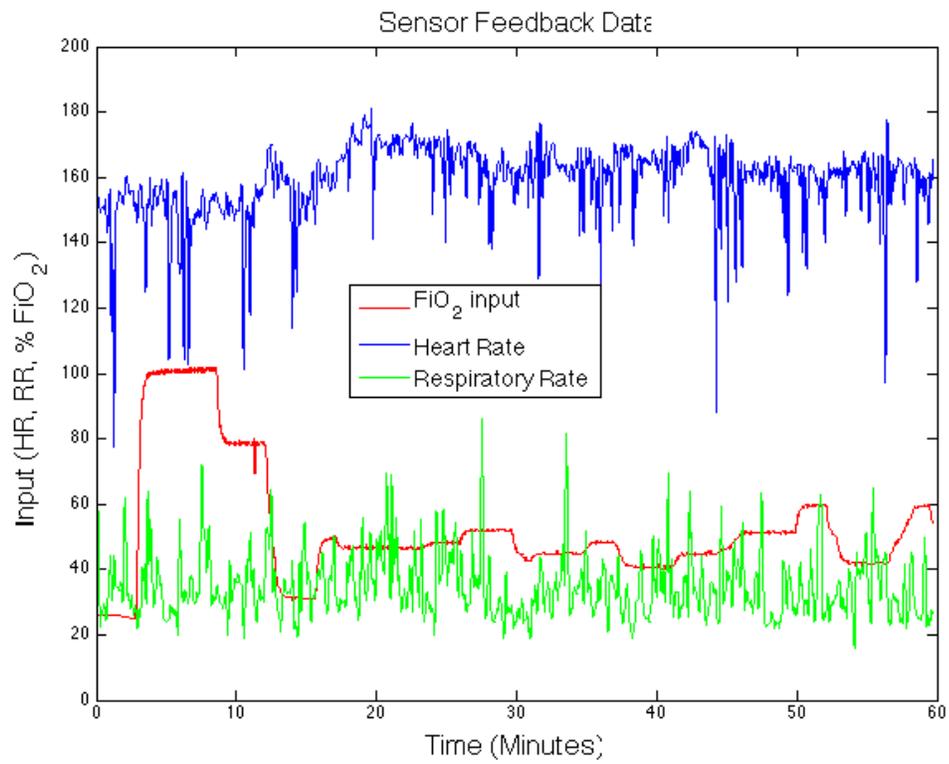


Fig. 24. Input data from clinical setting used in GA testing

4.3 Closed loop simulations

The goal of this adaptive parameter estimation technique is to ultimately select a controller that will effectively control the SpO_2 of the patient. It has already been demonstrated that the algorithm can select the correct controller and that the controllers meet the performance criteria. Here it will all be put together to demonstrate the selection and closed loop control at the same time.

Disturbance input data was derived from actual patient recordings as before. The FiO_2 was initially taken from patient readings as well but after the first iteration of the genetic algorithm a controller is selected which dictates the FiO_2 setpoint on the fly. Model parameters were given which fall under controller number 4 and for the entirety of the test the GA did select number 4. The parameter estimation can be seen in Fig. 25. More importantly the closed loop control history is shown in Fig. 26. where the values shown are normalized so that the desired control set point and the nominal FiO_2 are zero. The controller sets the FiO_2 and successfully maintains the SpO_2 within 1% despite the disturbance inputs. Notice the smooth adjustments in FiO_2 that limit overshoot. This control simulation confirms the operation of the controller according to the specifications and its ability to track a small steady state error.

The same test was done with parameters that fall within each nominal plant. In all cases a similar result was seen. When automatic control takes over in these simulations the SpO_2 is regulated as desired.

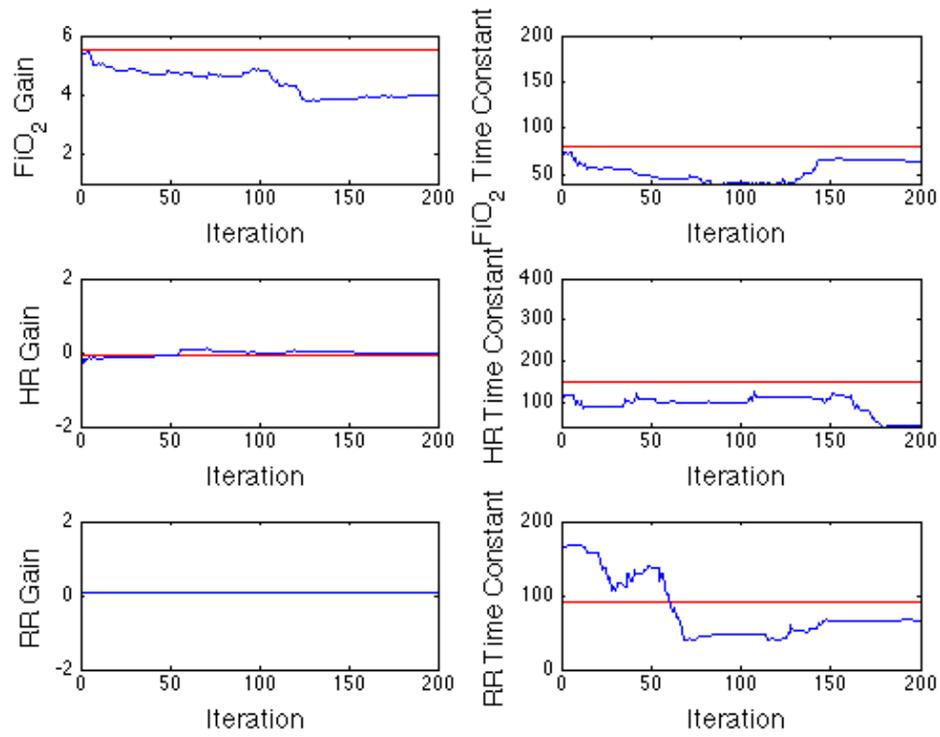


Fig. 25. Closed loop parameter estimation

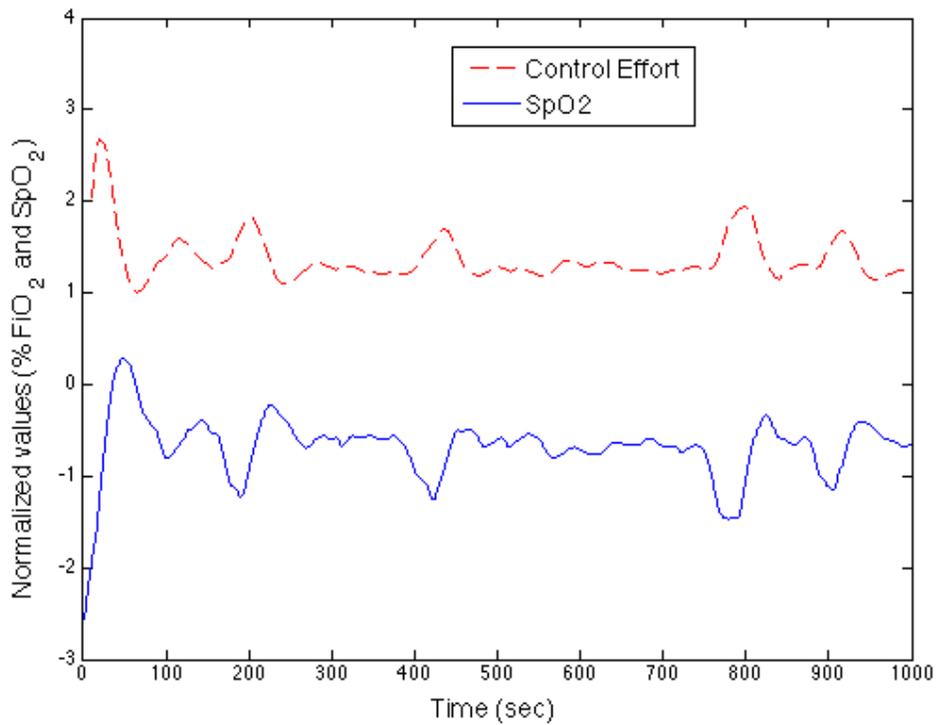


Fig. 26. Closed loop control test – SpO₂ and control effort

A second closed loop simulation was done for a longer time period to illustrate the controller's performance and ability to maintain a healthy SpO_2 with adverse disturbance inputs. The control effort, or FiO_2 , and the SpO_2 response can be seen in Fig. 27. The corresponding disturbances which are creating the small changes in SpO_2 can be seen in Fig. 28. It should be noted that the resolution of the feedback SpO_2 in these simulations is not commensurate with the existing sensor technology implemented in the hospital. These simulations provide high resolution data because they are based on calculations. The minute changes in SpO_2 seen in Fig. 27. would not be seen in a clinical setting and the control effort would not respond to those small changes as minutely.

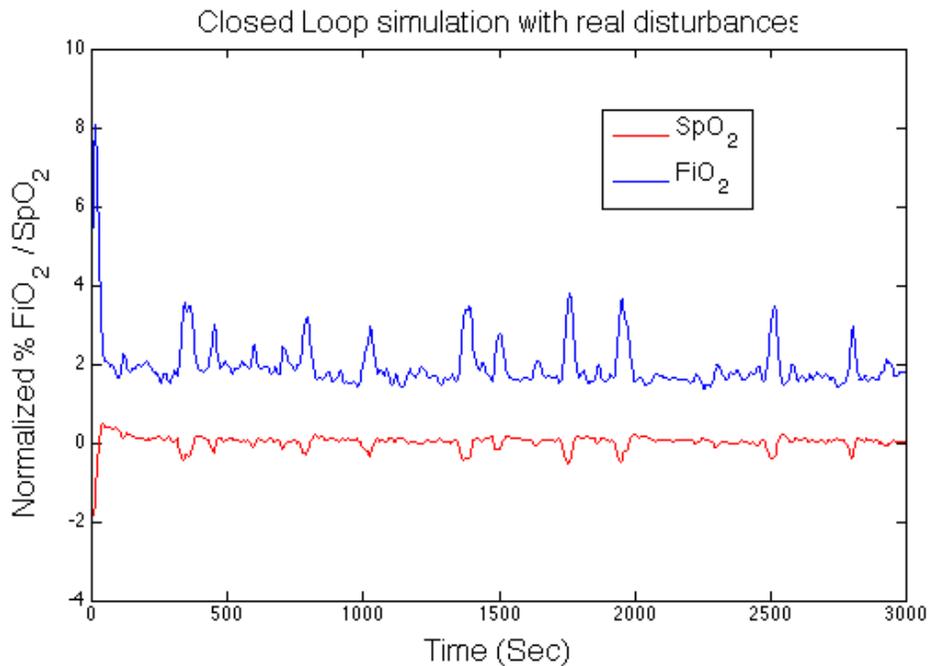


Fig. 27. Modeled closed loop results with real disturbance inputs from data

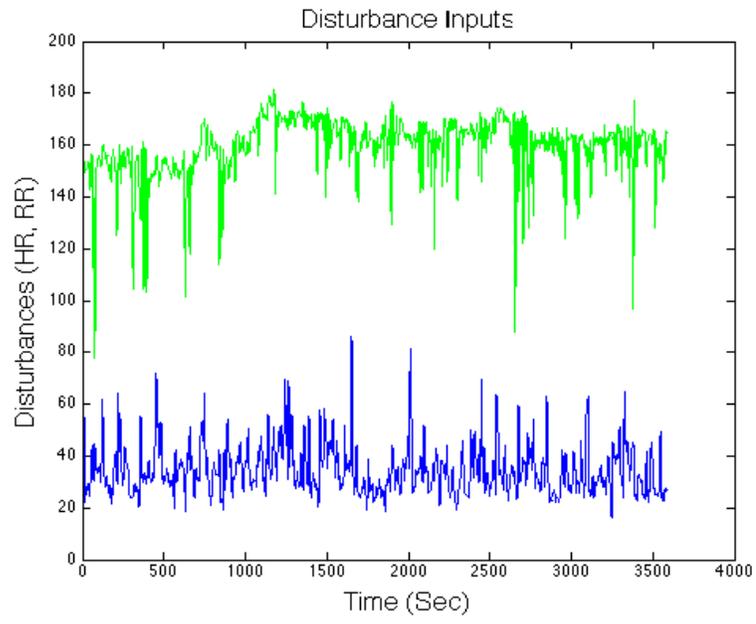


Fig. 28. Real disturbance input data used

Chapter 5: Prototype design and testing

A prototype device for carrying out automatic control of the FiO_2 in a clinical setting was needed. Previous clinical trials have not published designs for their means of controlling FiO_2 , only describing their necessary attributes. Presented here is an upgraded design for retrofitting a standard manual FiO_2 control valve into one that safely accepts automatic control. The design consists of an IV stand for mounting, a blend valve which is to be manipulated, a motor for control, a mounting system, and a microcontroller for accepting sensor feedback and making calculations. An assembly of these various parts is shown in Fig. 29. to help illustrate the components described below.

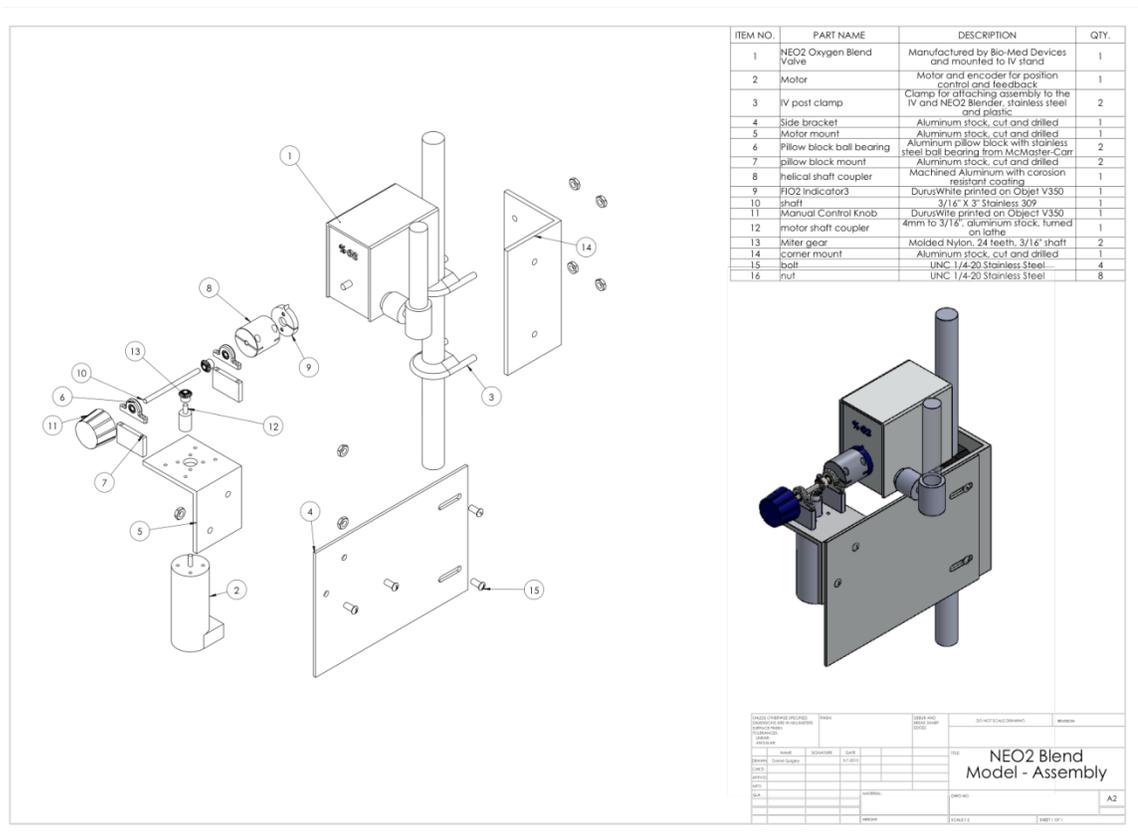


Fig. 29. NEO2 Blend valve retrofit design assembly drawing.

5.1 Device Performance Specifications

The performance specifications of this device are intended to match the needs of easy operation in the hospital, safety for the patient, and simplicity of design for reliability. Specifications for the overall size of the device were outlined by University Women's and Children's Hospital and easily met. As a whole it could not be bulky, protrude far beyond the face of the blender, and it must function similarly to the unmodified product. It was also mandated that the functionality of the blender or other devices not be obstructed in any way, meaning manual control must be possible at all times.

The overall concept of retro-fitting an existing O₂ mixing device eliminates unnecessary complexity and provides the same level of accuracy, reliability, and safety of the current industry standard in terms of the gas mixing mechanism. Since these devices are already in place in every NICU, there is no need to “re-invent the wheel” by designing a new O₂ mixer. The modification to the blend valve was influenced by the need to safely attach a motor for automatic control without hindering the normal operation by the user.

Operation of the device mimics the original blender in almost every way. There is a toggle switch for when manual control is desired, and a sensing element is in place so that the automatic control shuts off when a manual input is detected. The custom knob and indicator dial were designed to mimic the original appearance and operation of the blend valve prior to modifications. With the motor tucked safely away from the operator, the manual control knob and indicator dial allow normal operation of the blend valve when automatic control is turned off. Even when the device is shutdown and unpowered,

the manual control aspect of the blend valve exists exactly as before the modifications were made. This is important for the safety of the patient as well as for ease of use by the attending nurse or doctor.

When automatic control is turned on, an indicator light is turned on to assure that the electronics are functioning properly. Whenever the microcontroller box is powered on but automatic control is not enabled due to operator choice or malfunction, this indicator light flashes to show an error. These lights are located on the front panel of the microcontroller housing in Fig. 30.

Other performance specifications include the ability for the electronics to effectively communicate with the pulse oximeter, oxygen analyzer, and Spacelabs system to get proper feedback. This is done with serial cables, and communication with the attached laptop is possible with an Ethernet connection.

It was also specified that a calibration procedure be in place to ensure accurate feedback and O₂ control by the motor. In order to achieve this, a dialogue box provides instructions and forces the user to calibrate the device upon startup. There is also the option to re-calibrate during use if it is observed that the FiO₂ read by an oxygen analyzer does not match the input command.

The computational capabilities and computer hardware of the device have already been designed Krone and Keim and were not altered. The hardware and software allows feedback from FiO₂, SpO₂, HR, and RR sensors, which are connected via serial cable. It has the ability to control two motors so that a flow control algorithm can be added. It also has a vibration motor to physically stimulate the patient if HR or RR becomes irregular [23].

Some novel programming improvements have been made to account for the new physical aspects of the device. New wiring was added to the FPGA module to support the automatic control switch and lighting. The new DC servo motor hardware also required changes in the software. Control logic that will power off the motor when a manual intervention is recognized was added. Upgrades to the user interface for easier operation and recording of data were completed.

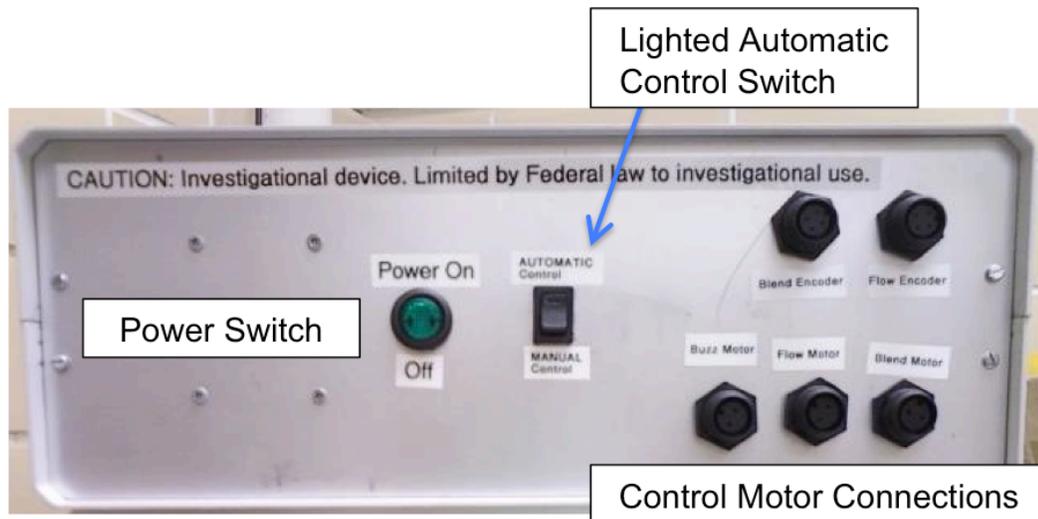


Fig. 30. Control panel and hardware box

5.2 Drawings and Final Design

Krone and Keim completed the first iteration of the prototype. This design achieved the desired ability to automatically control the FiO_2 by retrofitting the NEO2 blend value but did not meet size or safety and operation requirements. This original design can be seen in Fig. 29. The drawbacks included a lack of easy manual control, bulky overall size, a large exposed belt system, and a motor gearbox that did not turn freely when unpowered. The motor chosen, the belt system, and lack of knob meant there was no manual control override system that is crucial to limiting risk to the patient.

These limitations were revised in the second iteration of the prototype. The distance that the device protrudes from the face of the blend valve was reduced by $\frac{1}{2}$. The large motor and high friction gearbox were replaced with a smaller one with no gearbox that allows it to rotate freely when unpowered. A miter gear system replaced the belt drive to accommodate a manual control knob. A replacement indicator for the dial face of the blend valve was 3D printed along with the new manual control knob that mimics the original hardware. The improved design can be seen in Fig. 32.

The design was fabricated with aluminum panels, cut and drilled to mount the motor shaft. Pillow blocks were used to support the shaft. A nylon miter gear system transmits the motor's power. A helical beam shaft coupler attached the shaft to the blend valve and allows for small misalignments during operation. Detailed engineering drawings for these parts are illustrated in Appendix C.

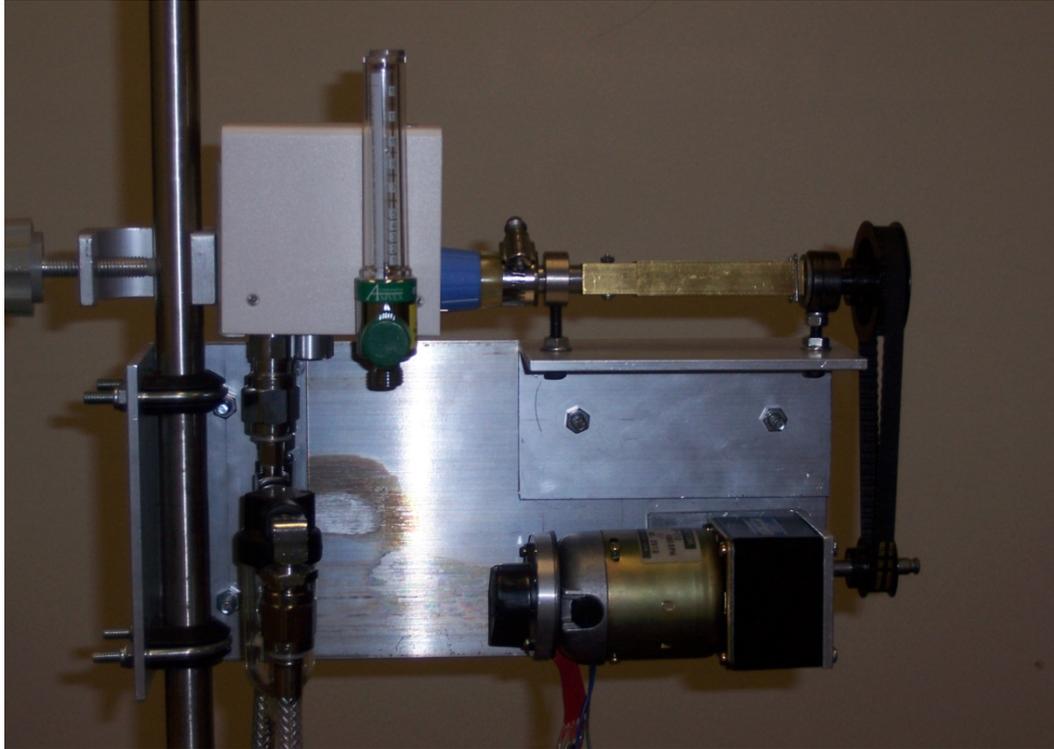


Fig. 31. Original prototype design, circa 2011



Fig. 32. New prototype design completed

Chapter 6: Discussion and Conclusion

6.1 Discussion

6.1.1 Open loop controller selection

The results of the genetic algorithm parameter estimator are very promising for the simulations that were done. Even though the algorithm uses a degree of randomness to introduce mutations and mate sets of parameters, the results such as those in Fig. 12 to Fig. 21 can be repeated with the same level of accuracy. The genetic algorithm displays the ability to quickly and consistently estimate appropriate parameters with enough accuracy to choose the correct nominal plant.

There is room for improvement in terms of limiting the mutation of the parameters once a certain level of error is reached. Occasionally a set of parameters is found that is incredibly close to the correct parameters but the need to continually search for better options, avoid local minimums, and adapt to potentially changing parameters makes it difficult for the GA to settle on a good set of parameters for very long. An example of this is in Fig. 12. through Fig. 14. near the end of the simulation where momentarily the error increases as incorrect parameters are chosen for a few iterations of the algorithm. This is definitely a drawback that could be addressed but typically at the expense of the other aspects of the parameter estimation scheme. It is a compromise between these aspects so some fine-tuning would be needed and actual clinical results would be very helpful in determining which aspects are most important; consistent parameter estimation or the ability to quickly adapt to new parameters.

Not only do the simulations show an ability to select one controller but also to adapt to changing parameters as in Fig. 16. through Fig. 21. It was shown that the GA can

adjust to increases or decreases in both the FiO_2 gains and time constants to select any of the six nominal plants. This is crucial to the long-term operation of the control system when being applied to a patient. Clinical trials would only require 24 to 72 hour operation but future implementation of the device as standard NICU equipment would potentially require weeks of continuous operation. Hour long simulations and worst case scenarios like the changing parameters in Fig. 19 can only do so much to prove the efficacy of the device, even with real data.

When the patient's actual model parameters lie very close between two plants the estimator exhibits some oscillations like in Fig. 15. This is a limitation of the system but there are ways of avoiding such occurrences. Since the operational range of the robust controllers overlap either one could be selected and would function properly in this case. An averaging system is also possible to avoid repeated switching between plants but would allow for gradual change of controllers if one plant was selected consistently over time as opposed to updating which controller is used upon every 5 second feedback interval.

The algorithm was also applied to raw data that was collected in a clinical setting as in Fig. 22 and Fig. . The accuracy of these results cannot be verified since they are technically unknown. It is encouraging, however, that the outputs of these tests appear to be very similar to the tests executed with known parameters. The parameter estimation is able to settle on consistent parameters with minor oscillations, occasional drifting, and then returning to the consistent parameters.

With both raw data simulations and simulations with modeled SpO_2 there is a problem with the FiO_2 parameter estimation when the FiO_2 input does not fully excite the

transfer function model. In all of the simulations done the FiO_2 inputs used were from sections of data where the FiO_2 varied frequently and in significant steps. When strings of data were used with constant FiO_2 values the parameter estimation for the disturbance inputs was accurate but the FiO_2 parameter estimation was not. This is justifiable because sections of data where the FiO_2 is nearly constant the SpO_2 remains in acceptable ranges are of no interest when trying to model the FiO_2 - SpO_2 relationship. In such cases the FiO_2 setting is already acceptable and changing the control system is unnecessary.

Another challenge that is inherent in the three transfer function model is the normalizing of the data and models. Since transfer function outputs are considered perturbations from nominal it is important to define the nominal value with which to normalize the data. Should the SpO_2 be normalized around the desired set point or updated each time, should the FiO_2 be normalized around 21% O_2 or some desired steady state value, are both difficult questions to answer. The model parameters that the GA chooses can be very different when the same data is analyzed with different nominal values being used.

A previous approach was to normalize all inputs about the first data point in the window being looked at. The window would update every 5 seconds and define new nominal values. The problem with this approach is the inputs at the beginning of the window could be very large inputs having a big effect on the SpO_2 but the GA views them as zero and not affecting the SpO_2 . With this approach the correct parameters will actually give a very bad fitness and be chosen against. This means the estimation system is actually driven away from the correct set of parameters. Another approach was to run the GA over the past hour of data in hopes that initial condition issues would disappear

over the hour, and then only analyze the most recent minute or 5 minute window. This approach is not very useable as a real time solution because of its data heavy approach that is not capable of completing every 5 seconds.

The solution utilized in these results was chosen for simplicity. It stores transfer function outputs and uses the percent of each transfer function's output compared to the SpO₂ to define an initial condition for the next updated window. This eliminates the need to normalize any data and minimizes processor requirements. The results shown clearly highlight the genetic algorithms ability to estimate the model parameters.

6.1.2 Closed loop controller selection

The results of the closed loop controller tests shown in Fig. 25 and Fig. helped confirm what was already known about the close loop system. The GA parameter estimator operates and chooses a controller, as it has been show to do in Fig. 12 through Fig. 21. The controllers are able to manage the SpO₂ despite the disturbance inputs, as they have been shown to do by the robust performance analysis described in Equations (10) - (12) and verified in Table 3 and Fig. 11.

The genetic algorithm was shown to run on a quad core laptop using parallel computing techniques at speeds that allowed real-time controller selection. This is to say that if feedback is given by the sensors in an NICU every 5 seconds the GA is capable of analyzing this data in under 5 seconds. This is very important because the work here is not intended to be purely theoretical. The system may not be implanted on the same hardware it was tested on but this shows that real-time operation is possible with the correct hardware. The closed loop simulations helped prove that the system could be

implemented in a clinical setting for further testing and refinement of the parameter estimation technique.

6.2 Recommendations

When analyzing and defining the control problem it was taken for granted that whatever adaptive control technique was selected would need to update real-time every 5 seconds. The sensor feedback gave new information on this interval so it was assumed that the analysis should be done on the same interval. However, upon more complete understanding of the physiological aspects of the patient it would be possible to reduce the frequency of the parameter estimation and still be able to adapt to changing parameters.

The changing of the infant's physiological properties occurs over hours or days not seconds. It would be possible to update the window less frequently, or do so selectively when adverse conditions occur such as a hypoxic or hyperoxic event.

A limited approach such as that could help solve one the main concern that was highlighted by the analysis done to obtain the results presented here: full excitation of the FiO_2 transfer function. The issue with the transfer function being fully excited especially applies to the closed loop tests because the goal of the controller is to maintain SpO_2 without exciting the FiO_2 input excessively. Since this is the goal, it is seen as a good problem to have.

A simple condition could be added to halt the GA analysis when FiO_2 has been set at a constant level for some time, and then continue the parameter estimation upon the next desaturation event. Such an option could save the processor from constantly being taxed highly by the GA as a secondary benefit. The exact implementation of this design idea was not pursued due to time constraints.

One final recommendation to improve this adaptive approach to robust control is an improved understanding of the accuracy of the GA. If a study was done to quantify the accuracy of the GA, that accuracy could be used to better define a series of robust controllers. It would be conceivable to compute a robust controller during real-time operation, which are defined by uncertainty ranges based on the accuracy of the GA. In this way the 6 predefined controllers could be replaced by an infinite number of unique controllers that would operate on a robust range that is always changing.

6.3 Conclusion

The objectives of this work were to incorporate an adaptive parameter estimating technique with the security of robust control in order to improve the care and health of the target patients. The automatic control was designed to mimic or possibly surpass the clinically proven best methods for manual control. The genetic algorithm method of model parameter estimation was to be improved to run accurately in real-time. These theoretical works were to be coupled with a prototype that could carry out the automatic control task safely.

On all accounts these goals were achieved. The genetic algorithm method of parameter estimation was improved and shown to consistently select the best set of parameters within real-time constraints. The system of 6 robust controllers was proven to manage the FiO_2 in a manner consistent with the defined performance specifications, which are rooted in clinical research. The latest iteration of the prototype has added safety measures and an improved user interface that will allow clinical research of the control system to ensue.

This has added to the literature an approach that will continue to improve health care for premature infants in need of respiratory support. The prototype design is ready for clinical testing of the control system and will provide a blueprint for further improvements. This method should continue to yield lower incidences of ROP, as with the studies described in the literature review, while reducing clinician workload. The integration of engineering research and best clinical practices as inspiration for innovation is especially important in the healthcare field.

Appendix A: Model information

Table 4 Uncertainty Model (W_A) for each of the six plants

Plant 1	$\frac{1.045 s^3 + 0.01645 s^2 + 0.0001307 s + 5.609e-07}{s^3 + 0.02212 s^2 + 0.0001735 s + 8.866e-07}$	(13)
Plant 2	$\frac{0.9815 s^3 + 0.0242 s^2 + 0.0003654 s + 3.394e-06}{s^3 + 0.02671 s^2 + 0.0003924 s + 3.854e-06}$	(14)
Plant 3	$\frac{0.8929 s + 0.004256}{s + 0.004844}$	(15)
Plant 4	$\frac{1.316 s + 0.007702}{s + 0.01654}$	(16)
Plant 5	$\frac{1.316 s + 0.007702}{s + 0.01654}$	(17)
Plant 6	$\frac{1.316 s + 0.007702}{s + 0.01654}$	(18)

Table 5 Uncertainty Model for Heart Rate and Respiratory Rate

HR	$\frac{30.26 s^2 + 0.2263 s + 0.0001988}{s^2 + 0.008735 s + 7.89e-06}$	(19)
RR	$\frac{31.4 s^2 + 34.96 s + 0.2783}{s^2 + 1.11 s + 0.01147}$	(20)

Appendix B: Controller Results

Table 6 Results of controller synthesis (K)

Plant 1	$\frac{-10.52 s^2 + 1.053e13 s + 1.131e11}{s^3 + 1.343e13 s^2 + 3.101e12 s + 1.256e09}$	(21)
Plant 2	$\frac{-8.759 s^2 + 8.78e12 s + 6.216e10}{s^3 + 9.338e12 s^2 + 1.967e12 s + 6.109e08}$	(22)
Plant 3	$\frac{-7.767 s^2 + 7.785e12 s + 4.087e10}{s^3 + 7.482e12 s^2 + 1.571e12 s + 3.906e08}$	(23)
Plant 4	$\frac{-10.26 s^2 + 1.027e13 s + 1.254e11}{s^3 + 4.072e13 s^2 + 9.602e12 s + 5.264e09}$	(24)
Plant 5	$\frac{-8.011 s^2 + 8.03e12 s + 5.98e10}{s^3 + 2.702e13 s^2 + 5.621e12 s + 2.022e09}$	(25)
Plant 6	$\frac{-6.438 s^2 + 6.443e12 s + 3.854e10}{s^3 + 2.128e13 s^2 + 3.94e12 s + 1.124e09}$	(26)

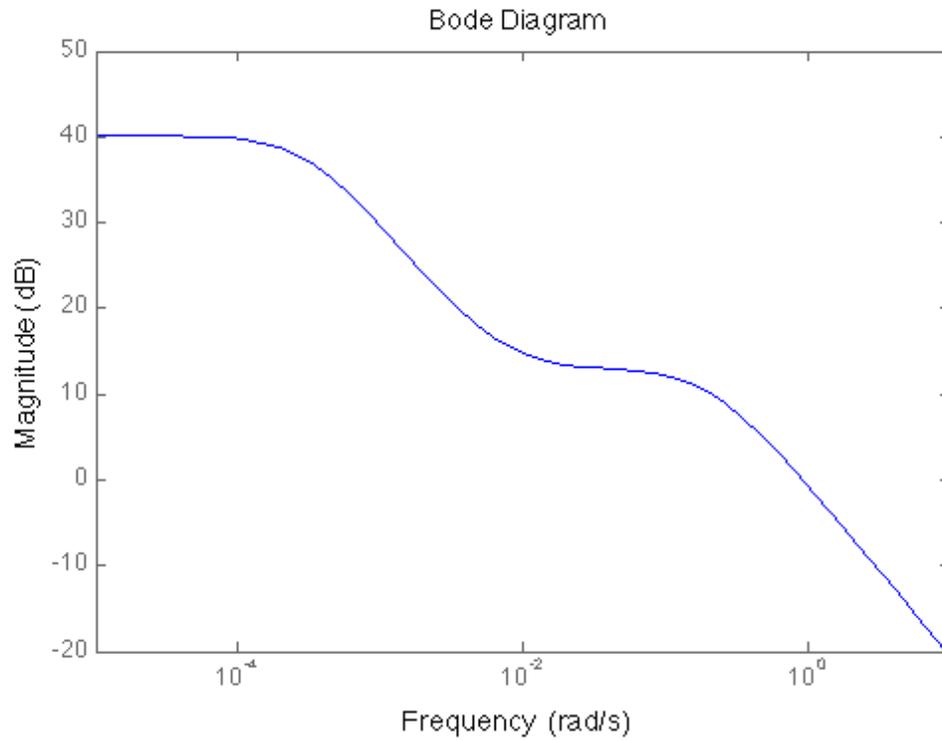
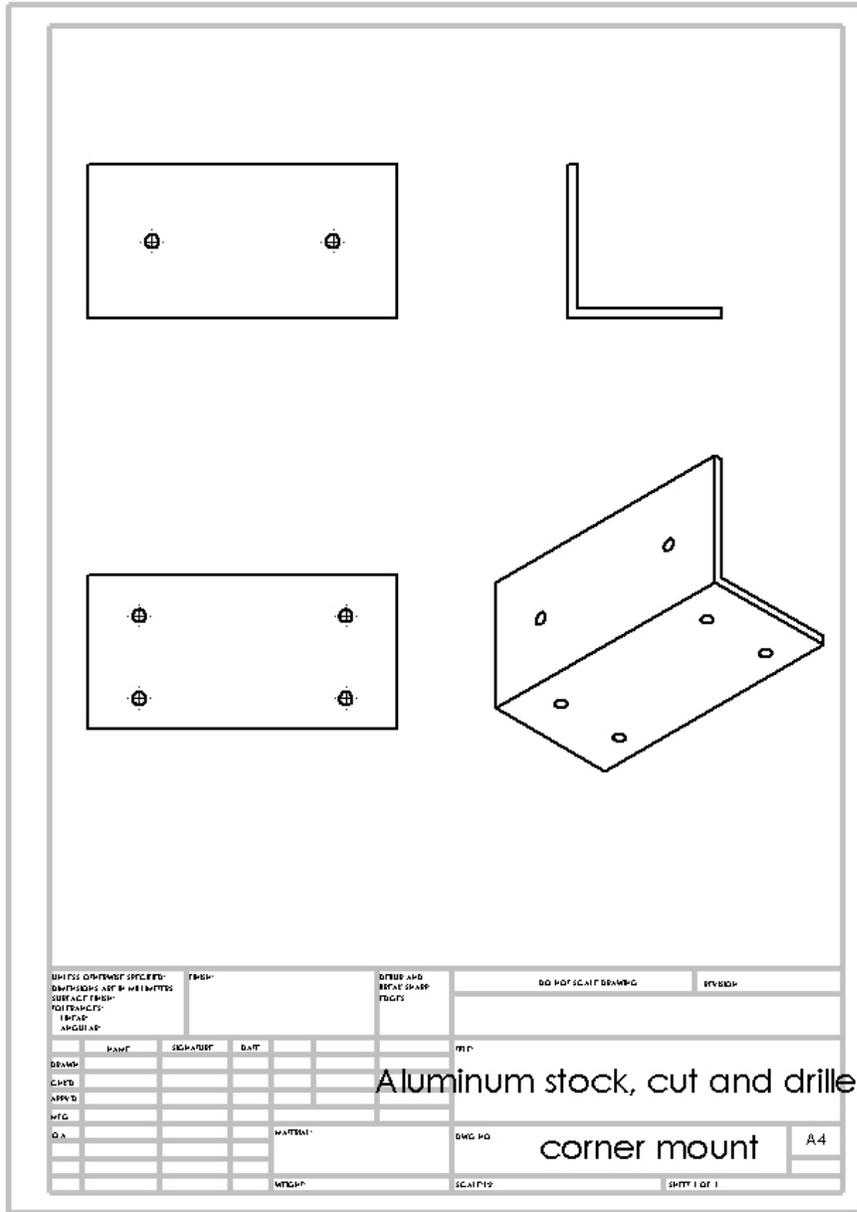
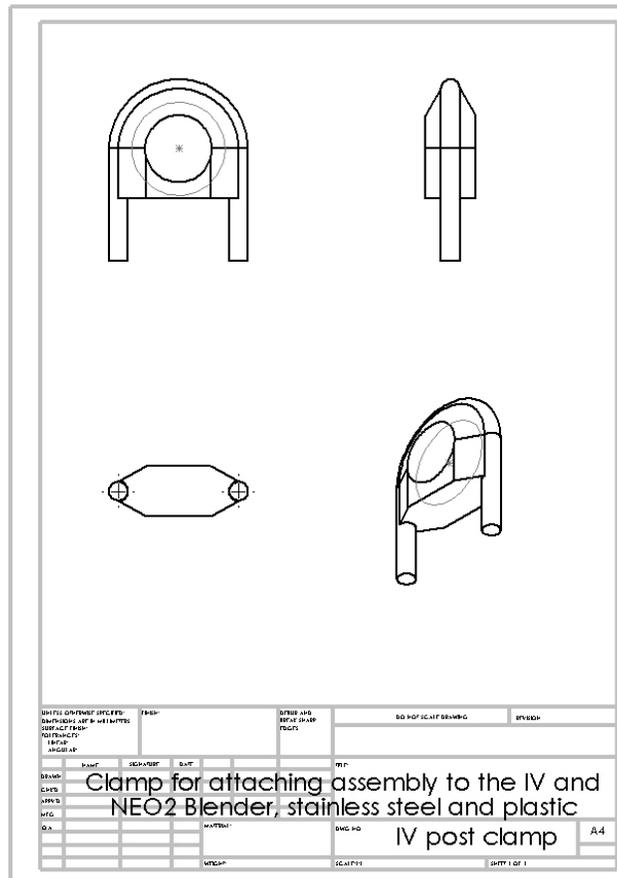
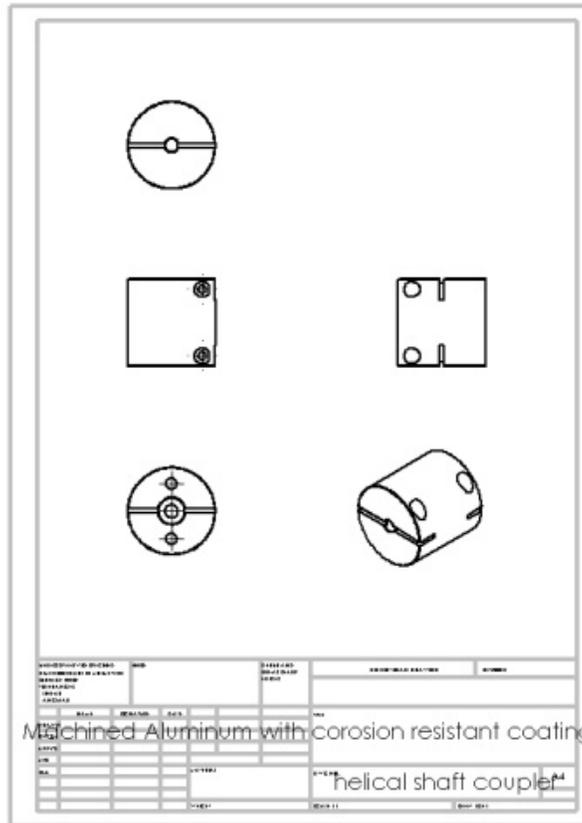
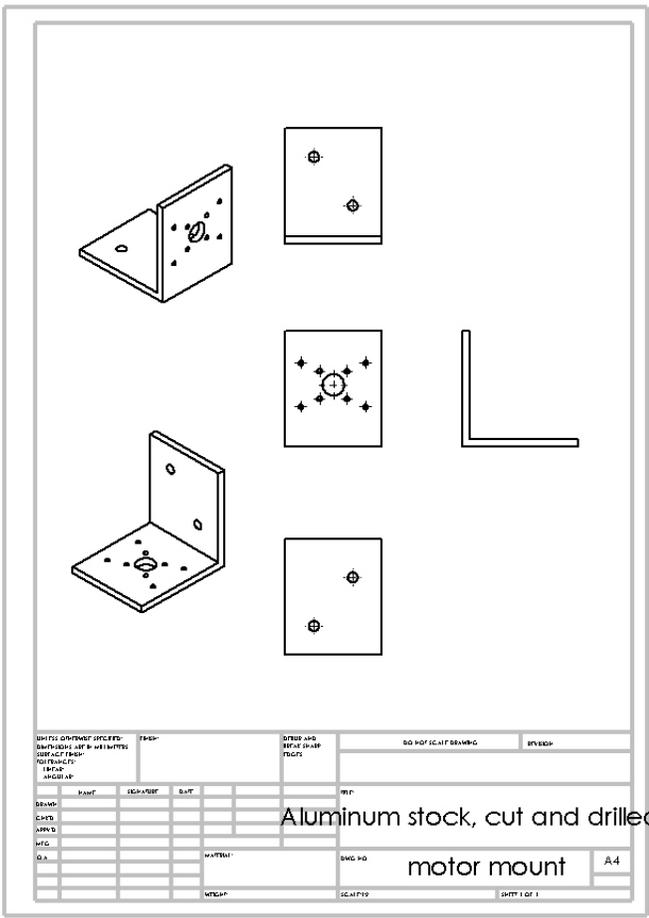
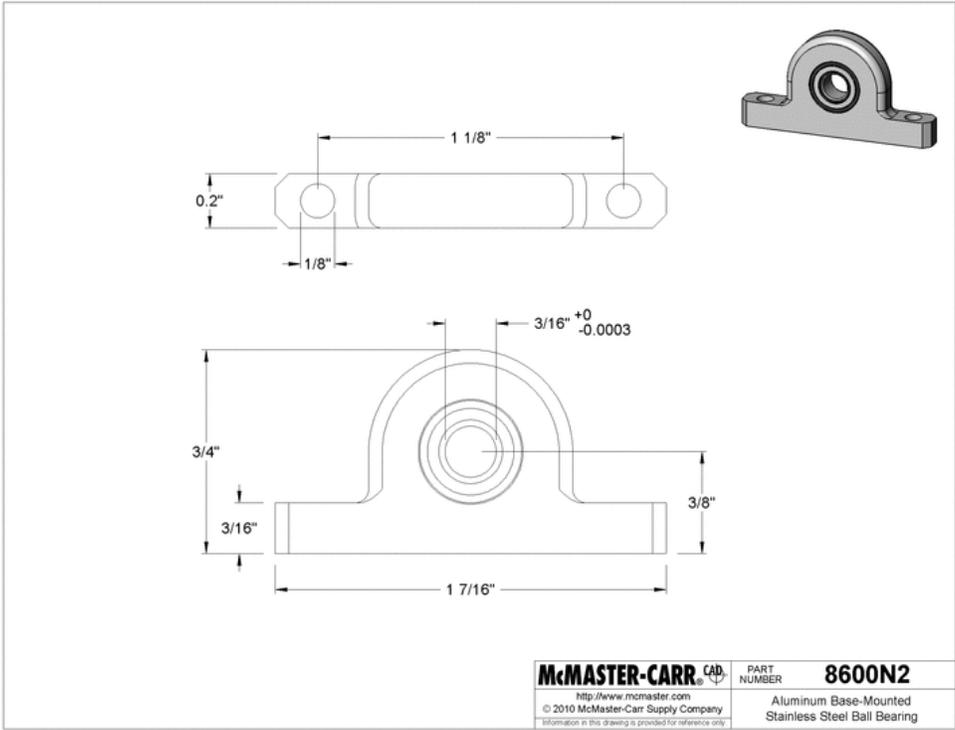


Fig. 33. Bode magnitude response of controller $K(j\omega)$ for nominal plant 2

Appendix C: Prototype Drawings







Appendix D: Genetic Algorithm MATLAB code

Performance verification of controllers:

```
function performance_check(flag)
load('W_HR_mult');W_HR=tf(W_HR_mult);
load('W_RR_mult');W_RR=tf(W_RR_mult);
load('WI_6controllers');
load('K_final2');

if flag==1
G_FiO2=2;      %gain for FiO2 TF
T_FiO2=95;    %time constant for FiO2 TF
elseif flag==2
    G_FiO2=2;T_FiO2=145;
elseif flag==3
    G_FiO2=2;T_FiO2=195;
elseif flag==4
    G_FiO2=5.5;T_FiO2=95;
elseif flag==5
    G_FiO2=5.5;T_FiO2=145;
elseif flag==6
    G_FiO2=5.5;T_FiO2=195;
end
G_FIO2=tf(G_FiO2,[T_FiO2 1]);
W_FIO2=tf(WI6{flag});
G_HR=tf(-.073,[177.2 1]);
G_RR=tf(-.09,[151.3 1]);
deriv=tf([1],[1 0]);

% define performance specifications
Wu_dot=0;%1/10;
Wu=1/79;
A=0.1;          %Low Frequency Error =10%
M=8.0;         %High Frequency Error =800%
w_b=2*pi*[1/160 1/160 1/210 1/110 1/160 1/210];
Wp=tf([1/M w_b(flag)],[1 w_b(flag)*A]); %wp=(s/M+w_b)/(s+A*w_b)

% construct P matrix
P=[0 0 0 0 0 0 W_FIO2;...
    0 0 0 0 W_RR*G_RR 0 0;...
    0 0 0 W_HR*G_HR 0 0 0;...
    -Wp*G_FIO2 -Wp -Wp -Wp*G_HR -Wp*G_RR 0 -Wp*G_FIO2;...
    0 0 0 0 0 0 Wu;...
    0 0 0 0 0 0 Wu_dot*deriv;...
    G_FIO2 1 1 G_HR G_RR 0 G_FIO2;...
    -G_FIO2 -1 -1 -G_HR -G_RR 1 -G_FIO2];

% construct partitions and N matrix
P11=P(1:7,1:6);P12=P(1:7,7);P21=P(8,1:6);P22=P(8,7);
N=P11+P12*K(flag)*(inv(1-P22*K(flag)))*P21;
N11=N(1:6,1:5);N12=N(1:6,6);N21=N(7,1:5);N22=N(7,6);
```

```

% nominal performance
np=sigma(N22);
NP=max(max(np));
%robust stability
rs=sigma(N11);
RS=max(max(rs));
%robust performance
rp=sigma(N);
RP=max(max(rp));

if NP<=1
    fprintf('\r nominal performance met N22=%2.2f \r',NP)
else fprintf('\r nominal performance NOT met N22=%2.2f \r',NP)
end
if RS<=1
    fprintf('\r Robust Stability met N11=%2.2f \r',RS)
else fprintf('\r Robust Stability NOT met N11=%2.2f \r',RS)
end
if RP<=1
    fprintf('\r Robust Performance met N=%2.2f \r',RP)
else fprintf('\r Robust Performance NOT met N=%2.2f \r',RP)
end

end

```

Uncertainty Model construction:

```

%Daniel Quigley
%Uncertainty calculations for HR & RR

%% Nominal models
sys_HR=tf(-.073,[177.2 1]);
sys_RR=tf(-.09,[151.3 1]);

%%
w=logspace(-4,2,1000);

[dummyHR]=freqresp(sys_HR,w);
[dummyRR]=freqresp(sys_RR,w);
for ii=1:max(size(dummyHR));sys_HR_nom(ii)=dummyHR(1,1,ii);end
for ii=1:max(size(dummyRR));sys_RR_nom(ii)=dummyRR(1,1,ii);end

for G_HR=-1.35:.2:1.2 %perturbation in gain of HR
for T_HR=101.17:10:253.3 %perturbation in time constant of HR
    sys_HR_p=tf(G_HR,[T_HR 1]);
    [dummy]=freqresp(sys_HR_p,w);
for ii=1:max(size(dummy));sys_p(ii)=dummy(1,1,ii);end%get a simple
vector from frequency response
    error_mag_mult=abs(sys_p-sys_HR_nom)./abs(sys_HR_nom);
%multiplicative error
    error_mag_add =abs(sys_p-sys_HR_nom);
%additive error
    figure(1);
    semilogx(w,20*log10(error_mag_mult),':');hold on
    figure(2);

```

```

        semilogx(w,error_mag_add, ':'); hold on
end
end

[dummyHR]=freqresp(sys_HR,w);
[dummyRR]=freqresp(sys_RR,w);
for ii=1:max(size(dummyHR));sys_HR_nom(ii)=dummyHR(1,1,ii);end
for ii=1:max(size(dummyRR));sys_RR_nom(ii)=dummyRR(1,1,ii);end

for G_RR=-1.466:.2:1.29 %perturbation in gain of HR
for T_RR=68:10:234.7 %perturbation in time constant of HR
    sys_RR_p=tf(G_RR,[T_RR 1]);
    [dummy]=freqresp(sys_RR_p,w);
for ii=1:max(size(dummy));sys_p(ii)=dummy(1,1,ii);end%get a simple
vector from frequency response
    error_mag_mult=abs(sys_p-sys_RR_nom)./abs(sys_RR_nom);
%multiplicative error
    error_mag_add =abs(sys_p-sys_RR_nom);
%additive error
    figure(3);
    semilogx(w,20*log10(error_mag_mult), ':');hold on
    figure(4);
    semilogx(w,error_mag_add, ':'); hold on
end
end
figure(1);xlabel('frequency (rad/sec)');ylabel('magnitude of error
1(\omega) (dB)');title('HR multiplicative error model');
figure(2);xlabel('frequency (rad/sec)');ylabel('magnitude of error
1(\omega) (dB)');title('HR additive error model');
figure(3);xlabel('frequency (rad/sec)');ylabel('magnitude of error
1(\omega) (dB)');title('RR multiplicative error model');
figure(4);xlabel('frequency (rad/sec)');ylabel('magnitude of error
1(\omega) (dB)');title('RR additive error model');
%% pick points to create W_HR_mult
figure(1)
% % pick 15 points from the error data on the screen (only the
magnitude data to be chosen)
errorData=ginput(15);
magError=(errorData(:,2));
freqr=errorData(:,1);%*2*pi;
magData=vpck(magError,freqr);% format the magnitude data for fitmag
%weight for fitmag (see fitmag help in MATLAB)
weightData=vpck(ones(size(freqr)),freqr);
% Obtain WI
figure(201);clf;
WI=fitmag(magData,weightData);
% Convert to state space
[A,B,C,D]=unpck(WI);
W_HR_mult=ss(A,B,C,D);
% Plot the bode plot of WI
figure(202);
bode(W_HR_mult,w);%,'r',{0.01,100});

[dummy,phase_error_model_I]=bode(W_HR_mult,w);
clear mag_error_model_HR_mult

```

```

for ii=1:max(size(dummy));mag_error_model_HR_mult(ii)=dummy(1,1,ii);end
figure(1)
hold on
semilogx(w, (mag_error_model_HR_mult), 'r')%make sure that the error
model bounds all of the error
save('W_HR_mult.mat', 'W_HR_mult')
% you may want to save your model so you can use it in another program
%% pick points to create W_HR_add
figure(2)
% % pick 15 points from the error data on the screen (only the
magnitude data to be chosen)
errorData=ginput(15);
magError=(errorData(:,2));
freqr=errorData(:,1);%*2*pi;
magData=vpck(magError,freqr);% format the magnitude data for fitmag
%weight for fitmag (see fitmag help in MATLAB)
weightData=vpck(ones(size(freqr)),freqr);
% Obtain WI
figure(201);clf;
WI=fitmag(magData,weightData);
% Convert to state space
[A,B,C,D]=unpck(WI);
W_HR_add=ss(A,B,C,D);
% Plot the bode plot of WI
figure(203);
bode(W_HR_add,w);%,'r',{0.01,100});

[dummy,phase_error_model_I]=bode(W_HR_add,w);
clear mag_error_model_HR_add
for ii=1:max(size(dummy));mag_error_model_HR_mult(ii)=dummy(1,1,ii);end
figure(2)
hold on
semilogx(w, (mag_error_model_HR_mult), 'r')%make sure that the error
model bounds all of the error
save('W_HR_add.mat', 'W_HR_add')
% you may want to save your model so you can use it in another program
%% pick points to create W_RR_mult
figure(3)
% % pick 15 points from the error data on the screen (only the
magnitude data to be chosen)
errorData=ginput(15);
magError=(errorData(:,2));
freqr=errorData(:,1);%*2*pi;
magData=vpck(magError,freqr);% format the magnitude data for fitmag
%weight for fitmag (see fitmag help in MATLAB)
weightData=vpck(ones(size(freqr)),freqr);
% Obtain WI
figure(201);clf;
WI=fitmag(magData,weightData);
% Convert to state space
[A,B,C,D]=unpck(WI);
W_RR_mult=ss(A,B,C,D);
% Plot the bode plot of WI
figure(202);
bode(W_RR_mult,w);%,'r',{0.01,100});

```

```

[dummy,phase_error_model_I]=bode(W_RR_mult,w);
clear mag_error_model_RR_mult
for ii=1:max(size(dummy));mag_error_model_RR_mult(ii)=dummy(1,1,ii);end
figure(3)
hold on
semilogx(w,(mag_error_model_RR_mult),'r')%make sure that the error
model bounds all of the error
save('W_RR_mult.mat','W_RR_mult')
% you may want to save your model so you can use it in another program
%% pick points to create W_RR_mult
figure(4)
% % pick 15 points from the error data on the screen (only the
magnitude data to be chosen)
errorData=ginput(15);
magError=(errorData(:,2));
freqr=errorData(:,1);%*2*pi;
magData=vpck(magError,freqr);% format the magnitude data for fitmag
%weight for fitmag (see fitmag help in MATLAB)
weightData=vpck(ones(size(freqr)),freqr);
% Obtain WI
figure(201);clf;
WI=fitmag(magData,weightData);
% Convert to state space
[A,B,C,D]=unpck(WI);
W_RR_add=ss(A,B,C,D);
% Plot the bode plot of WI
figure(203);
bode(W_RR_add,w);%,'r',{0.01,100});

[dummy,phase_error_model_I]=bode(W_RR_add,w);
clear mag_error_model_RR_add
for ii=1:max(size(dummy));mag_error_model_RR_mult(ii)=dummy(1,1,ii);end
figure(4)
hold on
semilogx(w,(mag_error_model_RR_mult),'r')%make sure that the error
model bounds all of the error
save('W_RR_add.mat','W_RR_add')
% you may want to save your model so you can use it in another program

```

References

- [1] Gerald B Merenstein, S. L. G., et al, 2011, Merenstein & Gardner's handbook of neonatal intensive care, Mosby Elsevier, St. Louis, MO.
- [2] Rodriguez RJ, M. R., and Fanaroff, 2002, "Respiratory distress syndrome and its management.," Neonatal-perinatal medicine: Diseases of the fetus and infant, F. a. Martin, ed., Mosby, St. Louis.
- [3] Wilson JL, L. S., Howard PJ, 1942, "Respiration of premature infants: response to variations of oxygen and increased carbon dioxide in inspired air," Am J Dis Child, 63, p. 1080:1085.
- [4] King, L., 2010, "Interview with Stevie Wonder," CNN LARRY KING LIVE.
- [5] Kumar, V., 2007, "Chapter 29: Eye, Retina and Vitreous, Retinal Vascular Disease," Robbins basic pathology (8th ed.), Saunders/Elsevier, Philadelphia.
- [6] Salyer, J. W., "Neonatal and Pediatric Pulse Oximetry," Proc. 31st RESPIRATORY CARE Journal Conference.
- [7] Lilly C. Chow, K. W. W., Augusto Sola, 2003, "Can Changes in Clinical Practice Decrease the Incidence of Severe Retinopathy of Prematurity in Very Low Birth Weight Infants?," Pediatrics, 111(2), pp. 339-345.
- [8] Hagadorn JI, F. A., Nghiem TH, Schmid CH, Phelps DL, Pillers DA, Cole CH, 2006, "Achieved versus intended pulse oximeter saturation in infants born less than 28 weeks gestation: the AVIOx study," Pediatrics, 118(4), pp. 1574-1582.
- [9] A Laptook, S., W, Allen, J, Saha, S, Walsh M, 2006, "Pulse Oximetry in Very Low Birth Weight Infants: Can Oxygen Saturation be Maintained in the Desired Range?," Perinatol, pp. 337-341.
- [10] L. Clucas, D., LW, Dawson, J, Donath, S, Davis, PG, 2007, "Compliance with Alarm Limits for Pulse Oximetry in Very Preterm Infants," Pediatrics, 119, pp. 1056-1060.
- [11] Tehrani, F. T., and Bazar, A. R., "An Automatic Control System For Oxygen Therapy Of Newborn Infants," Proc. Engineering in Medicine and Biology Society, 1991. Vol.13: 1991., Proceedings of the Annual International Conference of the IEEE, pp. 2180-2182.
- [12] Bradley Krone, R. F., Ramak Amjad, 2011, "Model of neonatal infant blood oxygen saturation," Dynamic Systems and Control ConferenceArlington, VA.

- [13] Beddis, I., Collins, P, Levy, NM, Godfrey, S, Silverman, M, 1979, "New Technique for Servo-Control of Arterial Oxygen Tension in Preterm Infants," *Arch Dis Child*, 54, pp. 278-280.
- [14] Dugdale R E, C. R. G., Lealman G T, 1988, "Closed-loop control of the partial pressure of arterial oxygen in neonates," *Clin. Phys. Physiol. Meas.*, 9(4), pp. 291-305.
- [15] Morozoff, P. E., Evans, R. W., and Smyth, J. A., "Automatic control of blood oxygen saturation in premature infants," *Proc. Control Applications, 1993.*, Second IEEE Conference on, pp. 415-419 vol.411.
- [16] Claire, N., Gerhardt, T, Everett, R, Musante, G, Herrera, C, Bancalari, E, 2001, "Closed-Loop Controlled Inspired Oxygen Concentration for Mechanically Ventilated Very Low Birth Weight Infants with Frequent Episodes of Hypoxemia," *Pediatrics*, 107, pp. 1120-1124.
- [17] Miksch S, S. A., Horn W, Popow C, 1999, "Abstracting Steady Qualitative Descriptions over Time from Noisy, High-Frequency Data," *Proceedings of the Joint European Conference on Artificial Intelligence in Medicine and Medical Decision Making*, Springer, pp. 281–290.
- [18] Urschitz, M., Horn, W, Seyfang, A, Hallenberger, A, Herberts, T, Miksch, S, Popow, C, Muller-Hansen, I, Poets, C, 2004, "Automatic Control of the Inspired Oxygen Fraction in Preterm Infants: A Randomized Crossover Trial," *American Journal of Respiratory and Critical Care Medicine*, 179, pp. 1095-1100.
- [19] Morozoff E.P, S. J. A., Evans R.W., "Automatic SaO2 control using adaptive modelling," *Proc. Canadian Medical and Biological Engineering Society conference*, p. 144.
- [20] Morozoff, P. E. S., J. A., 2009, "Evaluation of Three Automatic Therapy Control Algorithms on Ventilated Low Birth Weight Neonates," 31st Annual IEEE EMBS Minneapolis, MN, pp. 3079-3082.
- [21] Claire N, D. U. C., Bancalari E, 2009, "Automated Adjustment of Inspired Oxygen in Preterm Infants with Frequent Fluctuations in Oxygenation: A Pilot Clinical Trial," *The Journal of Pediatrics*, 155(5), pp. 640-646.
- [22] Claire N, B. E., D-Ugard C, Nelin L, Stein M, Ramanathan R, Hernandez R, Donn S, Becker M, Bachman T, 2011, "Multicenter Corssover Study of Automated Control of Inspired Oxygen in Ventilated Preterm Infants," *Pediatrics*, 127(1), pp. 76-83.
- [23] Krone, B., 2011, "Modeling and Control od Arterial Oxygen Saturation in Premature Infants," *Masters in Science*, University of Missouri, Columbia, MO.

- [24] Keim, T., 2011, "Control of Arterial Oxygen Saturation in Premature Infants," Doctor of Philosophy, University of Missouri, Columbia, MO.
- [25] Kenneth W Wright, D. S., Lisa Thompson, Rangasamy Ramanathan, Roy Joseph, Sonal Farazavandi, 2006, "A physiologic reduced oxygen protocol decreases the incidence of threshold retinopathy of prematurity," *Trans Am Ophthalmol Soc*, 104, pp. 78-84.
- [26] R. Maurice Hood, A. D. B., Alfred T. Culliford, Garay S, Kamelar D, 1989, "Pathophysiology of trauma-associated respiratory failure," *Thoracic Trauma*, W.B. Saunders Company.
- [27] Fraser, R. G., 1988, *Diagnosis of Diseases of the Chest*, W B Saunders, Philadelphia.
- [28] Yu, C. H., W.G. ; So, James M. ; Roy, Rob ; Kaufman, H. ; Newell, Jonathan C. , 1987, "Improvement in Arterial Oxygen Control Using Multiple-Model Adaptive Control Procedures," *Biomedical Engineering, IEEE Transactions on, BME-34(8)*, pp. 567 - 574.
- [29] Timothy Keim, R. A., Roger Fales, "Modeling and control of the oxygen saturation in neonatal infants," *Proc. Dynamic Systems and Control Conference*.
- [30] Sigurd Skogestad, I. P., 2005, *Multivariable Feedback Control analysis and design*, John Wiley & Sons, West Sussex, England.