

Tissue Type Recognition for Laser-Based Cancer Treatment

Floris van Rossum^{1,2}, Kayla Swiston², Yixue Wang^{1,3}, and Irene Wong³

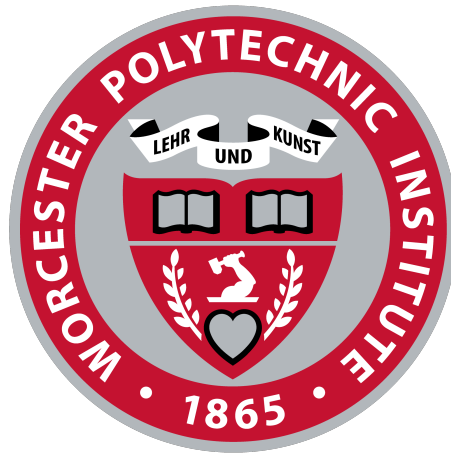
¹Department of Computer Science, Worcester Polytechnic Institute

²Robotics Engineering Program, Worcester Polytechnic Institute

³Department of Mathematical Sciences, Worcester Polytechnic Institute

May 2020

Project Advisors: Prof. Andrea Arnold³ and Prof. Loris Fichera^{1,2}



Abstract

This multidisciplinary (MA/CS/RBE) project aims to address concerns surrounding in-office laser surgeries in the throat. These procedures, also known as in-office laryngeal laser procedures, are a safer alternative to procedures conducted in an operating room. During in-office laser procedures, surgeons have difficulty in determining the amount of laser exposure necessary to successfully treat harmful cells while minimally damaging surrounding healthy tissue. This results in risks associated with in-office surgical procedures such as a possible need for re-treatment or complications arising from improper laser delivery. Through the research of laser-tissue interactions, we build mathematical models that estimate tissue parameters which allow us to classify the tissue according to its thermal response. The hypothesis that these tissue parameters change over time allows us to monitor the changes in tissue properties resulting from laser exposure. Knowledge of these tissue properties can then help to aid a surgeon in better determining the completion of a surgical laser procedure.

Executive Summary

Introduction

In-office surgical procedures, such as in-office laser treatment, use local anesthesia for the duration of the procedure [1]. By avoiding general anesthesia, these procedures provide a suitable option for vulnerable patients, such as young or elderly individuals or those who are unable to afford general anesthesia. However, current laryngeal laser procedures can result in complications or the need for re-treatment due to the lack of resources the surgeon has to determine the success of the laser delivery. The resulting complications and required re-treatments cost patients and insurance companies additional time and money. The goal of this research is to reduce the need for re-treatment and minimize complications following in-office laryngeal laser procedures by providing the surgeon with additional information about the state of tissue impacted by the surgical laser. We aim to accomplish this goal by classifying whether tissue has been sufficiently treated, and relaying this information to the surgeon so they can better judge the completion of their procedure.

Background

During an in-office laser procedure, topical anesthesia is applied to the larynx, and the laser fiber is brought to the target area with a flexible laryngoscope which includes channels for a camera and lighting [2]. The completion of the laryngeal procedure is currently based on visual changes, like color changes, in the target area observed by the surgeon. Surgeons look for a change from red to brown or from red to black to signify an end-point for laser exposure [2]. For pathologies located on the surface of the larynx, treatment is considered by observing tissue blanching, vaporization, and coagulation [2].

There are 5 types of laser-tissue interactions; each type is defined by specific tissue properties and laser parameters. An important parameter is the exposure time, which strongly determines the type of interaction that occurs. This paper focuses on thermal interactions and the tissue modifications that result from the various effects of this interaction. We are interested in studying the temperature profiles of the tissue surface, and determining whether the tissue is adequately or inadequately exposed to the laser during a laser procedure.

Methodology

We hypothesize that various tissue types have different properties and thermal responses. We use a Gaussian model to represent the temperature profile of the tissue. The parameters of the Gaussian model include the amplitude (A), standard deviation (σ), and center location (μ). Assuming that μ is constant, we define the A and σ as time-varying. With A and σ changing over time, we study the evolution of these parameters to assist us in approximating tissue properties and determining the tissue type.

By building an estimation framework we can predict the evolution of the time-varying parameters for both A and σ . This framework requires the application of the Bayesian statistics and the Ensemble Kalman Filter (EnKF). Particularly, we are interested in the joint posterior density, which defines the density of the parameters given the observed temperature data. To verify our estimation framework, we generated various sets of simulated data. With the assumption that the tissue surface has even properties, we can define appropriate A and σ functions, in terms of time t , to present the evolution of A and σ during a laser-tissue interaction. We used multiple sets of simulated data to verify our EnKF framework under different conditions, including the cases of single-pulse and multiple-pulse laser application.

Using Linear Discriminant Analysis (LDA) and Support Vector Machine (SVM) classification models, we created a framework for classifying tissues based on estimated parameters from the mathematical models

described in the previous paragraph. Through exploratory data analysis, we identified derivations of the estimation sequences that would allow for classification of tissue types based on their thermal response. The final four classification features selected were the mean and maximum of the A and σ sequences. SVM and LDA models were trained on 1000 data point models and tested with 100 data points.

To further verify the mathematical framework, we built an experimental setup comprised of a Sharplan 30C CO₂ surgical laser to fire pulses at Agar phantom tissue, a FLIR A655sc Infrared Thermal camera to collect thermal data, and a Franka Emika PANDA robotic arm to support and maneuver the laser fiber. The experimental trials involved varying laser power settings in the range of 5 - 20 Watts, varying laser pulse duration in the range of 0.1 - 0.7 seconds, varying the ratio of Agar powder to water in the Agar solutions, and optionally including red food dye in the Agar samples. Data capture included both multiple pulse trials and single pulse trials. Work on the experimental data collection was halted due to campus closure from COVID-19.

Results

After verifying the parameter estimation framework with sets of simulated data for both single pulse and multiple pulses, we made observations on the effects of various initial inputs on the accuracy of the resulting estimations. These inputs include the type of distribution for the initial parameters, the observation error, the model prediction error, the standard deviation of the random walk, the ensemble size, and the grid size during data pre-processing. Our observational conclusions were made from the common results of multiple test runs of the estimation framework, factoring in the accuracy of the resulting estimations as well as the execution time of the framework. We concluded that a uniform distribution (with the endpoints of 0 and 100) for the initial A and σ , with an ensemble size of 500, would produce better estimations promptly. In terms of error, we observed that a lower error value will result in better estimations, but a value that is too low would be unrealistic. To make this balance, we concluded on desired ranges of value for both the observation error and model prediction error. In terms of the standard deviation of the random walk, we perceived that a lower value better captures smaller changes in the parameters and larger values capture greater changes over time. Lastly, for data pre-processing, we concluded that a grid size of 20 results in more accurate estimations in a reasonable time. Overall, these observational conclusions are essential in improving the accuracy and duration of the EnKF framework. The LDA and SVM classification models trained were successful in correctly identifying simulated thermal spreads based on the outputs of the estimation framework. The classification model is able to distinguish thermal responses with a difference in amplitude of 30 °C.

The reliability and repeatability of the preliminary data collection experiments are unclear, as the documentation and preparation processes for important control variables, such as height of the laser from the Agar gel, and starting temperature of the Agar gel before firing, were not consistently monitored, noted, nor replicated between experiments. Further work will have to be done to verify the findings of the mathematical model with experimental data. Regardless, the data analysis tool built to examine the experimental data is operational, which allows us to choose the observed temperature data and estimate its time-varying parameters. However, the analysis tool is still in its early stages. The future of this project requires a better design of the analysis tool that includes the classification module and feedback to the surgeon.

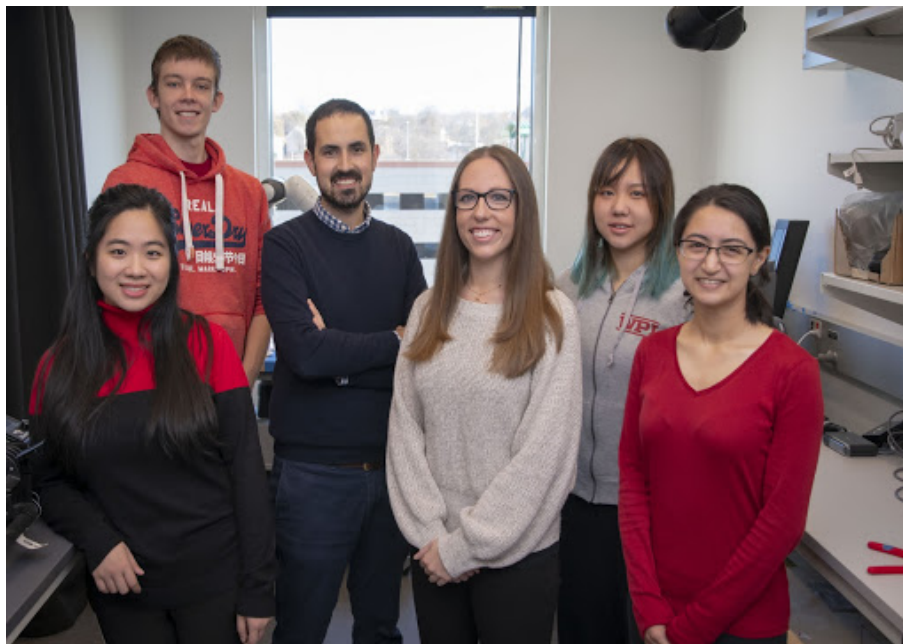
Conclusions and Future Work

This project proposes a parameter estimation framework, a thermal response classifier, an experimental setup for further thermal data collection, and an experimental analysis tool. The EnKF estimation framework is verified with the particular cases we have considered in our data simulation, like for a single pulse and multiple pulses. Additionally, from the tests we conducted with various framework settings, the framework can produce more accurate estimations in reasonable execution time. The LDA and SVM classification models were tested with simulated data points of assigned data types. Four features were identified to

allow for classification of the data sets: the mean and maximum of both the estimated amplitude and standard deviation sequences. There exist many options for the further development of the work done on the classification of thermal responses. Further study is needed to apply the results of this project to real-world surgical applications. The estimation framework will have to be further verified with experimental data.

Acknowledgments

We would like to acknowledge our advisors for this project, Professor Andrea Arnold and Professor Loris Fichera, for their hard work and dedication to the progress and completion of this project, and our continual improvement. We would like to acknowledge Worcester Polytechnic Institute (WPI) for providing us with access to laboratory facilities and equipment, as well as granting us financial support in order to make required purchases for this research project. We would also like to acknowledge the support of the National Science Foundation (NSF) grant number NSF/DMS-1819203. Finally, we would also like to acknowledge Dr. Thomas Carroll for providing us with medical knowledge on in-office laser treatments, and for his continual support and enthusiasm for this project.



This figure depicts our student research team and our project advisors at the research laboratory. From left to right: Irene Wong, Floris van Rossum, Professor Loris Fichera, Professor Andrea Arnold, Yixue Wang, Kayla Swiston.



NSF/DMS- 1819203

Contents

List of Figures	ix
List of Tables	xii
1 Introduction	1
2 Laryngeal Laser Procedure & Laser-Tissue Interactions	4
2.1 In-Office Laryngeal Laser Procedures	4
2.2 Laser-Tissue Interaction Types	5
3 Methodology	7
3.1 Mathematical Modeling	7
3.2 Computer Simulation of Sequential Temperature Data	9
3.3 State and Parameter Estimation Framework	11
3.3.1 Overview of the Nonlinear Filtering	12
3.3.2 Pre-Process the Discrete Temperature Sequence	12
3.3.3 Initialization of the Ensemble	13
3.3.4 Prediction Step	13
3.3.5 Observation Update Step	14
3.3.6 Outputs of the Estimation Framework	15
3.4 Machine Learning Models and Techniques for Classification of Thermal Data	16
3.4.1 Implementation of the Classifier	16
3.4.2 Simulated Data Generation	17
3.4.3 Classification Testing	18
4 Numerical Results	19
4.1 Data Simulation and Parameter Estimation: Single Pulse	19
4.2 Data Simulation and Parameter Estimation: Multiple Pulses	21
4.2.1 Additional Pulse is Delivered After the Cooling of Previous Pulse	21
4.2.2 Residual Heat	23
4.2.3 Change in the Center Location	26

4.3	Estimation Framework Settings with Simulated Data	28
4.3.1	Different Types of Initial Prior Distributions	28
4.3.2	Observation Error	29
4.3.3	Model Prediction Error	30
4.3.4	Parameter Random Walk Deviation	30
4.3.5	Input Variables of Uniform Distribution and the Ensemble Size	31
4.3.6	Choice of Grid Size in Data Pre-Processing	32
4.4	Classification Results	35
4.4.1	Amplitude Difference in Classification	35
4.4.2	Robustness of the Classification Model	35
5	Development of a Bench-Top Experimental Setup	41
5.1	Data Collection Materials and Methods	41
5.1.1	Considerations on the Design of the Experimental Setup	41
5.1.2	Equipment	42
5.2	Running Experiments to Collect Real Data	42
5.2.1	Tissue Targets	42
5.2.2	Experimental Protocol	43
5.3	Preliminary Data Collection	43
5.4	Experimental Data Analysis	44
6	Conclusions and Future Work	49
	References	51
	Appendix A Mass-Spring System: An Application of EnKF	53
A.1	State Estimation with Traditional Kalman Filter	53
A.2	State Estimation with EnKF	55
A.3	State and Parameter Estimation with EnKF	56
	Appendix B Results of Estimation Framework Settings	59
B.1	Distribution Types Testing	59

B.2	Observation Error Testing	62
B.3	Model Prediction Error Testing	65
B.4	Random Walk Testing	70
B.4.1	Results from a Single Pulse Simulation	70
B.4.2	Results from a Multiple Pulse Simulation	73
B.5	Inputs Variables of Uniform Distribution and Ensemble Size Testing	77
Appendix C Authorship Table		84

List of Figures

1	Diagram of Laryngeal Procedure	1
2	Our Approach	3
3	Olympus ENF-V2 Flexible Video Rhino Laryngoscope	4
4	User-Interface Mockup	5
5	Laser Interaction Types	6
6	Temperature Profile Captured by a Thermal Camera	8
7	Amplitude and Standard Deviation during Laser-Tissue Interaction	10
8	Amplitude and Standard Deviation during Heating and Cooling	11
9	Simulated Temperature Plot	11
10	Histograms of Random Draws for A and σ	13
11	Distinguishable Classifying Features	16
12	Maximum Amplitude for Simulated Tissue Types	17
13	Frames of a Simulated Temperature Sequence	19
14	Amplitude and Standard Deviation Functions Used in Simulating Data	20
15	Estimation of Parameters for a Single Pulse	20
16	Absolute Errors of the Amplitude and Standard Deviation Estimations	21
17	Amplitude and Standard Deviation Functions for Two Consecutive Laser Pulses	22
18	Estimation Results of the EnKF Framework for Two Consecutive Pulses	22
19	Absolute Errors of the Estimation Framework on Two Consecutive Pulses	23
20	Simulation of Residual Heat with Multiple Pulses	24
21	Simulation of Residual Heat with Multiple Pulses of Variable Length	24
22	Estimation Results of the EnKF Framework on Multiple Pulses with Residual Heat	25
23	Absolute Error of the EnKF Framework on Multiple Pulses with Residual Heat	25
24	Varying the Center of the Simulated Laser Pulse	27
25	Parameter Estimation Results for Multiple Pulses with a Changing Location of Application	27
26	Absolute Errors for Multiple Pulses with a Changing Location of Application	28
27	Histograms of Different Prior Distributions of Parameters	29
28	Estimation Model Performance with a Grid Size of 10	33

29	Estimation Model Performance with a Grid Size of 20	34
30	Estimation Model Performance with a Grid Size of 50	34
31	Difference in Maximum Amplitude in Classification	36
32	LDA Classification Results	37
33	SVM Classification Results	38
34	Amplitude Differences in Testing Data	39
35	Classification with Noise in the Maximum Amplitude	40
36	Classification with Noise Results	40
37	The Experimental Data Collection Setup	41
38	The Left and Middle Panels of the Experimental Analysis Application	45
39	The Right Panel of the Experimental Analysis Application	46
40	Position and Velocity State Estimations from the Traditional Kalman Filter	55
41	Position and Velocity State Estimations from the Ensemble Kalman Filter with Known Parameters	56
42	Position and Velocity State Estimations from the Ensemble Kalman Filter with Unknown Parameters	58
43	Damping Coefficient and Spring Constant Estimation from Ensemble Kalman Filter	58
44	Exponentially Distributed Values in Estimation Model	60
45	Weibull Distribution in the Estimation Model	60
46	Poisson Distribution in the Estimation Model	61
47	Normal Distribution in the Estimation Model	61
48	Uniform Distribution in the Estimation Model	62
49	Observation Error of 0.005 in Parameter Estimation ($\delta_D^2 = 0.005$)	63
50	Observation Error of 0.01 in Parameter Estimation ($\delta_D^2 = 0.01$)	63
51	Observation Error of 0.1 in Parameter Estimation ($\delta_D^2 = 0.1$)	64
52	Observation Error of 0.5 in Parameter Estimation ($\delta_D^2 = 0.5$)	64
53	Observation Error of 1 in Parameter Estimation ($\delta_D^2 = 1$)	65
54	Model Prediction Error Testing Where $\delta_C^2 = 0.0001$	66
55	Model Prediction Error Testing Where $\delta_C^2 = 0.001$	66
56	Model Prediction Error Testing Where $\delta_C^2 = 0.01$	67
57	Model Prediction Error Testing Where $\delta_C^2 = 0.05$	67

58	Model Prediction Error Testing Where $\delta_C^2 = 0.1$	68
59	Model Prediction Error Testing Where $\delta_C^2 = 0.5$	69
60	Model Prediction Error Testing Where $\delta_C^2 = 1$	69
61	Estimation Results for A Single Pulse with a Standard Deviation of 0.005	71
62	Estimation Results for A Single Pulse with a Standard Deviation of 0.01	71
63	Estimation Results for A Single Pulse with a Standard Deviation of 0.025	72
64	Estimation Results for A Single Pulse with a Standard Deviation of 0.05	72
65	Estimation Results for A Single Pulse with a Standard Deviation of 0.1	73
66	Estimation Results for Multiple Pulses with a Standard Deviation of 0.01	74
67	Estimation Results for Multiple Pulses with Standard Deviations of 0.01 (A, σ) and 0.001 (μ)	75
68	Estimation Results for Multiple Pulses with Standard Deviations of 0.01 (A, σ) and 0.02 (μ)	76
69	Input Variables of Uniform Distribution in Ensemble Size Testing: Test 1 Results	78
70	Input Variables of Uniform Distribution in Ensemble Size Testing: Test 2 Results	79
71	Input Variables of Uniform Distribution in Ensemble Size Testing: Test 3 Results	80
72	Input Variables of Uniform Distribution in Ensemble Size Testing: Test 3 Results of Less Accurate Estimates	81
73	Input Variables of Uniform Distribution in Ensemble Size Testing: Test 4 Results	82
74	Input Variables of Uniform Distribution in Ensemble Size Testing: Test 5 Results	83

List of Tables

1	Inputs and Units of the Heat Diffusion Equation	7
2	Units and Inputs of the Gaussian Function	9
3	Parameters of Simulated Data	17
4	Estimation Framework Parameters	18
5	Classification Experiments	18
6	Inputs to the Estimation Framework for Observation Tests	32
7	Execution Times for the EnKF Framework	32
8	Execution Time Table of Estimating Pre-Processed Temperature Data with Different Grid Size	33
9	Additional Table of Inputs of the Estimation Framework for Observation Tests	77

1 Introduction

In-office surgical procedures are a safer and more accessible option for the treatment of many medical conditions that would normally require treatment in an operating room. In-office surgery is generally less expensive than traditional surgery since patients do not incur operating room charges. Furthermore, in-office procedures can often be performed in conscious patients with the administration of partial (or topical) anesthetic as opposed to general anesthesia [1]. Therefore, in-office procedures are a viable option for patients who are particularly vulnerable such as the elderly [2].

Otolaryngology, i.e., the branch of medicine which focuses on the treatment of pathologies of the ear, nose, and throat, has recently seen the introduction of numerous novel office-based techniques. In this paper, we focus on laser laryngeal procedures, whose goal is to remove pathologies located on the larynx or vocal folds [3]. Laryngeal laser surgery is performed using the endoscopic approach illustrated in Figure 1, which shows a laryngoscope, carrying a camera and laser fiber, inserted into the nose and down the throat to visualize and treat a benign tumor on the larynx. Before the laser pulses are delivered to the tissue, topical anesthesia is applied. This form of anesthesia allows the patient to stay conscious and leave the office shortly after the completion of the procedure, instead of requiring them to recover in the hospital [4]. The unhealthy tissue is exposed to the laser until thermal damage occurs and appropriate physical changes, such as discoloration, in the tissue are apparent to the surgeon [2]. As in-office laser treatments have been shown to be safe and cost-effective, these procedures are becoming increasingly more accepted [5].

Despite their growing popularity, office-based laser laryngeal procedures remain challenging to perform [6]. A surgeon has no methods available to assist in controlling and applying the laser to the target area during a procedure, or to judge the laser’s effect. The ideal end-point of the procedure is when enough energy is applied to sufficiently treat the target tissue while minimizing damage to the surrounding healthy tissue. However, the end of the procedure is currently determined by the surgeon’s visual analysis of the target tissue [2]. With current methods, there is a risk of over-exposure of the laser, which results in thermal damage to surrounding healthy tissue and can lead to complications such as bleeding and swelling. These complications lead to discomfort and a longer recovery time for the patient [7]. Conversely, an insufficient amount of laser exposure on the targeted tissue will result in a recurrence of the problem and may require re-treatment. An assistive technological system that would allow the surgeon to determine the required pulses for sufficient treatment could improve the outcome of the procedure by reducing complications and the need for re-treatment.

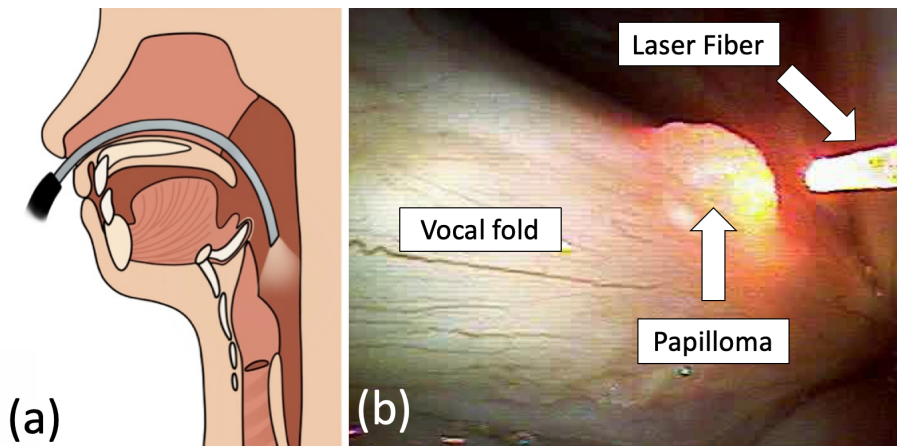


Figure 1: (a) An illustration representing a laryngeal procedure where a surgeon passes an laryngoscope through the nose to treat pathologies on the vocal folds. (b) A still from a camera close to the laser tip during a laryngeal procedure where a Papilloma is exposed to laser pulses.

The challenges of in-office laryngeal laser procedures are highlighted in a study performed by Del Signore et al. involving 255 patients [2]. Of these patients, 27% required a revision surgery due to insufficient treatment in the first procedure [2]. A study by Anderson et al. documented a number of complications [1]. These complications include vocal fold scar formation with resulting stiffness, vocal fold bowing, and loss of vibratory function which can result in unexpected dysphonia – difficulty in speaking as a result of a physical disorder of the vocal folds [2]. Some complications encountered, such as scar formation, can be treated with additional laser surgery while others resolve over time. These complications result from laser energy delivered to the surrounding healthy tissue and causing thermal damage, which can be avoided by providing the surgeon with a better understanding of the state of the target tissue.

Aiming to overcome the challenges described above, the long-term goal of this research is to reduce the need for re-treatment and minimize complications following in-office laryngeal laser procedures by providing the surgeon with additional information about the tissue impacted by the surgical laser. This extra information will allow the surgeon to minimize internal damage and maximize target removal. Using a mathematical model to estimate time-varying parameters relating to the tissue, we wish to develop a method to inform the surgeon about the state of the tissue. Using these tissue parameters, we aim to classify whether tissue has been sufficiently treated. This classification allows the surgeon to better determine whether a specific area was properly treated by the laser, thus minimizing the need for re-treatment and risk of collateral damage to healthy tissue.

In this paper, we lay the groundwork for the vision outlined above. We explore the use of a mathematical model with a Gaussian function to predict the surface temperature and temperature diffusion of the tissue after exposure to a laser pulse. This estimation can be used to provide a classification on the state of the tissue depending on its thermal response. The estimation and classification of the mathematical model can be verified through experiments with simulated data generated with known tissue parameters and laser settings. This simulated data is fed into the mathematical model, allowing it to predict time-varying parameters and classify the tissue as treated or untreated. The mathematical model can be visualized in real-time to provide direct feedback about the impacted tissue to a surgeon. This feedback will allow the surgeon to avoid areas classified by the model as sufficiently treated by the laser, reducing the thermal damage of healthy tissue. The block diagram in Figure 2 represents our approach described above.

The outline of this document is as follows: Chapter 2 covers the general background knowledge necessary for this project, such as information on laser-tissue interactions and mathematical modeling. In Chapter 3, we discuss our approach and methodology to build a mathematical model, collect experimental data, and create a tissue-type classifier. Chapter 4 discusses our numerical results including the simulated temperature data we generated with the mathematical model and different inputs, observational tests on the performance of the estimation framework and the results generated by the classifier. Chapter 5 covers the experimental setup developed for the collection of thermal data with the purpose of verification and analysis tool for experimentally collected data. Chapter 6 provides our conclusions and explores future steps in exploring this goal.

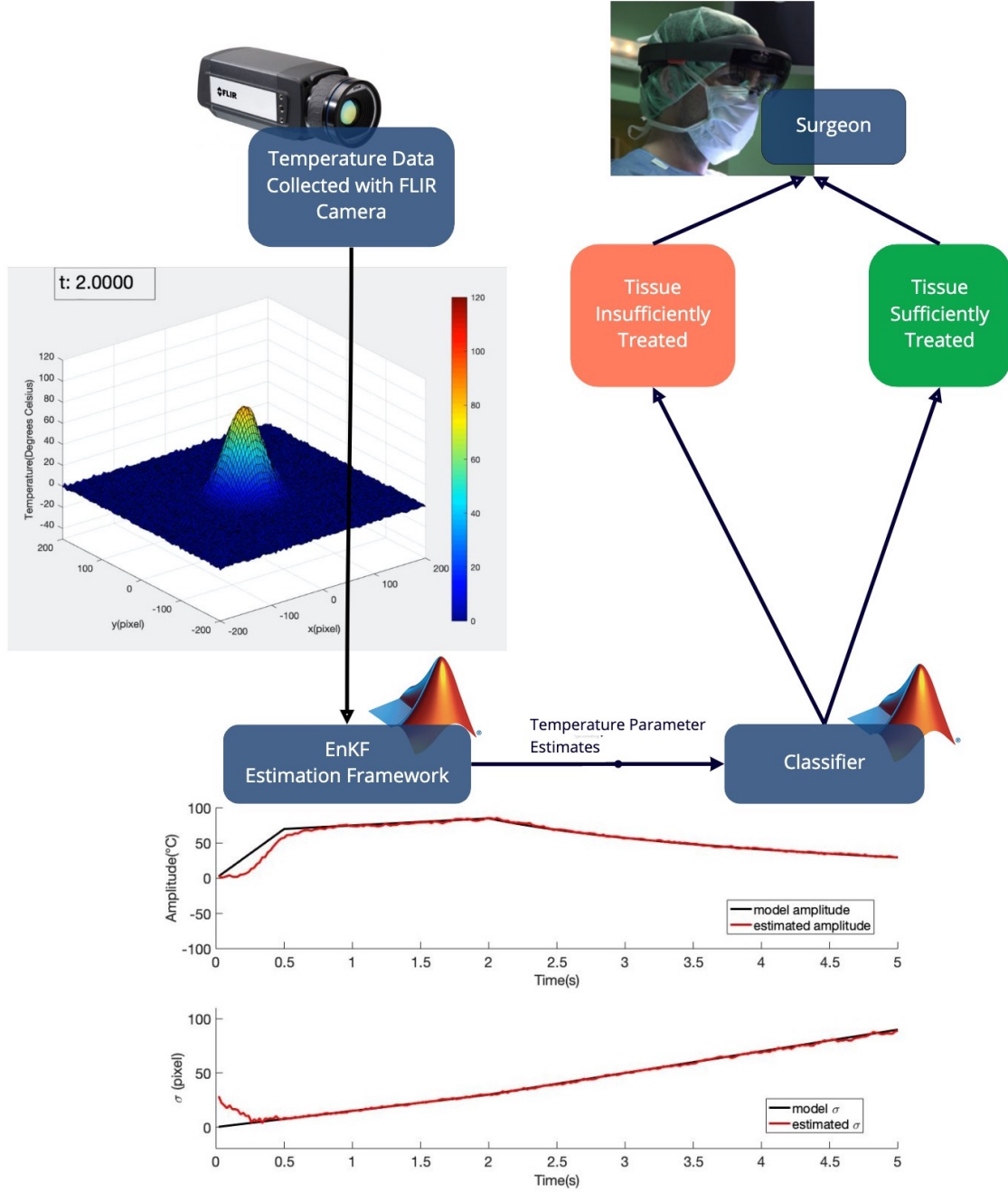


Figure 2: Schematic representation of our proposed tissue classification framework. Observed tissue temperature profiles are collected by a FLIR camera and are input into an estimation model which predicts parameter sequences relating to the data. These sequences are input into the classification model to obtain, identify, and distinguish the thermal response.

2 Laryngeal Laser Procedure & Laser-Tissue Interactions

2.1 In-Office Laryngeal Laser Procedures

Endoscopic laryngeal laser procedures are an example of in-office procedures whose goal is to treat pathologies located on or around the larynx. Examples of the pathologies that are good candidates for this approach include papillomas, granulomas, ectasias, varices, and polyps [2]. The laryngeal procedure represents a cheaper and more accessible alternative to other forms of laryngeal treatment, such as suspension laser microlaryngoscopy or transoral laser microsurgery [8]. These procedures are performed under general anesthesia in an operating room [8].

In-office laser treatments are typically performed with a potassium titanyl phosphate (KTP) laser with a wavelength of 532 nm due to its selective absorption in hemoglobin [5]. The properties of this laser cause a haemostatic effect and reduces energy transfer to surrounding healthy tissues [1]. This haemostatic effect, or the stoppage of blood flow, is critical during a procedure. Bleeding can obscure visualization of the target area, cause the energy of the laser not to be absorbed by the target unhealthy tissue, thereby minimizing the therapeutic effects of the laser [6]. Thanks to recent developments in flexible laser technology, the procedure can also be performed with a CO₂ laser of 10,600 nm wavelength [6, 9]. The CO₂ laser is largely absorbed by tissues with high water content and displays similar a haemostatic effect to the KTP laser [9].

The laser fiber is delivered to the target area with a flexible laryngoscope, as seen in Figure 3, which includes channels for a camera and lighting [2]. Prior to the insertion of the laryngoscope, topical anesthesia is applied to the larynx in the form of lidocaine with a drip catheter. Topical anesthesia is also applied to the nasal passages with xylometazoline and lidocaine spray [1, 2]. The application of lidocaine gives a 30-minute window to complete the procedure [6]. In the studies performed by Anderson et al., 16.6 % of procedures used pre-procedure sedation with lorazepam due to patient anxiety [1].

The decision to stop a laryngeal procedure is currently based on changes in appearance of the tissue. For pathologies located on the surface of the larynx, treatment is considered by observing tissue blanching, vaporization, and coagulation [2]. In prior studies, surgeons have expressed difficulty in maintaining consistent visualization of the target area and laser distance to the tissue, due to the mobility of the conscious patient [2, 6]. Specifically, the vocal fold and laryngoscope continuously move relative to each other making effective laser application challenging [2].

Due to the problems with laser control and observation of the target tissue, ensuring success in in-office laryngeal procedures is challenging. There exists a risk of complications and failures or the need for re-treatment. Overexposing the target tissue to the surgical laser results in thermal damage of surrounding healthy tissue. This damage can then result in long-lasting scars, which may need to be treated with additional procedures [2]. Meanwhile, insufficient use of the laser can result in an incomplete treatment,



Figure 3: An Olympus ENF-V2 Flexible Video Rhino Laryngoscope used in office-based laryngeal surgery [3]. Reproduced from [10].

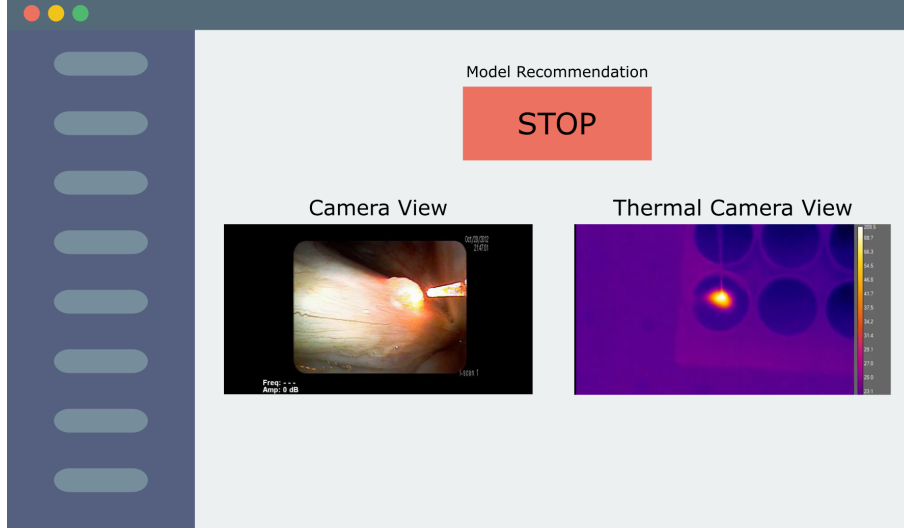


Figure 4: A User-Interface (UI) mock-up representing a tool to assist a surgeon during a laryngeal laser procedure. At the top is a label with a recommendation from the mathematical model to assist the surgeon in determining the end point of the procedure. The bottom two images allow the surgeon to visualize the state of the target tissue with a regular camera and thermal camera.

which means additional procedures may be required. Aiming to tackle these limitations, in this paper we propose a technology to help surgeons better understand and monitor the interactions between the laser and the tissue in an effort to minimize the damage to healthy tissue and maximize removal of target tissue. Figure 4 illustrates a concept of the technology we wish to propose. We envision a system that receives temperature information from a thermal camera pointed at the tissue, and that analyzes the temperature response to laser irradiation to intelligently estimate the state of the tissue. This information would be provided to a physician to assist in deciding when the procedure is completed. A possible configuration of the user-interface for this software is illustrated in Figure 4. The user interface would feature both a regular and thermal view from the tip of the laryngoscope, as well as parameters that display the effectiveness of the laser pulse and state of the tissue.

2.2 Laser-Tissue Interaction Types

Before we can formulate our approach, we first have to review the fundamental physical phenomena that occur when laser light and biological tissue interact. In this section, we provide a review of our current understanding of laser-tissue interactions.

Laser-tissue interactions are traditionally divided into five main categories, as depicted in Figure 5: photodisruption, photoablation, plasma-induced ablation, thermal interactions, and photochemical interactions. The occurrence of a certain interaction mechanism is determined by the specific properties of the tissue and the parameters of the laser. Tissue properties include, but are not limited to, optical parameters such as coefficients of reflection, absorption, and scattering, as well as thermal parameters such as heat conduction and heat capacity. Laser parameters include wavelength, exposure time, applied energy, focal size, energy density, and power density. An important setting is the exposure time, which is a major determining factor for the type of interaction that occurs. This paper focuses on thermal interactions and the tissue modifications that result from the various effects of this family of interactions, since these normally occur during surgical procedures [12].

Thermal interactions take place with an exposure time between $1\mu\text{s}$ and 1 minute for power densities between 10 W/cm^2 and 10^5 W/cm^2 . The delivery of laser pulses with these characteristics results in the

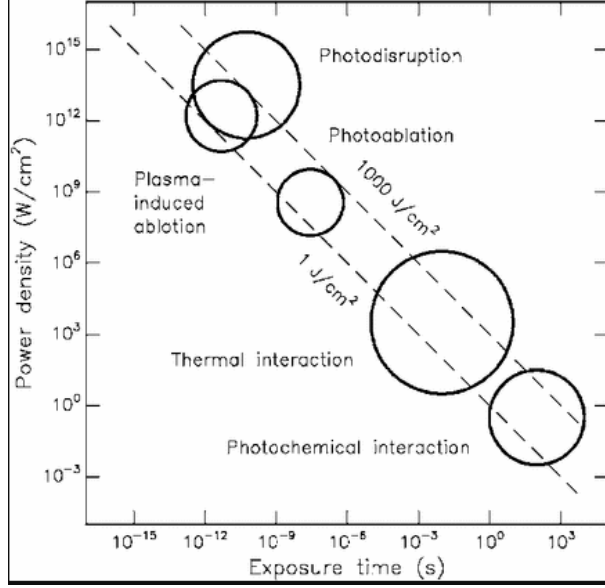


Figure 5: This graph depicts the five types of laser interaction with respect to power density and exposure time. Reproduced from [11].

creation of a local temperature increase. Depending on the temperature level, different effects can be observed in the tissue: coagulation, vaporization, carbonization, and melting. Coagulation occurs when the tissue temperature reaches above 60 °C and causes irreversible thermal injury to the tissue. Vaporization occurs when layers of the tissue with high water content absorb the laser energy, and the water in the tissue expands as it vaporizes [12]. When tissue temperature reaches above 100 °C, the tissue undergoes carbonization. Melting occurs when the tissue temperature reaches several hundred degrees Celsius.

During thermal interactions, changes occur to the visual appearance of the tissue. For example, the tissue becomes darker during coagulation and bubbles at 100 °C, at which water in the tissue arrives at boiling point. As described in Section 2.1, during in-office laryngeal laser treatments, the surgeon relies on visual appearance changes in tissue surface, like color, to determine whether the success of the treatment and when the targeted tissue is adequately exposed and treated by the laser. However, deciding proper endpoints with visual feedback alone is challenging to the surgeon. Estimating the amount of energy delivered to the tissue can also be challenging due to the energy delivery fluctuations caused by the movement of the patient during the procedure, as described in [2]. In this project, we are interested in studying the temperature profiles of the tissue surface and determining whether the tissue is adequately or inadequately exposed.

As heat is added to the tissue due to laser exposure, tissue properties, including the capacity to absorb and diffuse laser heat, change. We hypothesize that various types of tissue have different such properties and unique thermal response to the laser. If this hypothesis is valid, it would open up to the possibility of recognizing the tissue properties and its type, based on the observation of its thermal response to controlled laser irradiation. To understand the validity of our hypothesis, we need to study how lasers generate heat into tissue.

3 Methodology

3.1 Mathematical Modeling

Prior research has shown that the application of high-intensity laser light to biological tissue creates a local rise in temperature described by the following partial differential equation [12],

$$\frac{\partial T}{\partial t} = \frac{1}{\rho c} S + \kappa \nabla^2 T, \quad S = \alpha I \quad (1)$$

where T is the temperature of the tissue, t is time, ρ is the tissue density, c is the specific heat capacity, S is the volumetric power density created by the laser, κ is the tissue temperature conductivity, α is the absorption coefficient of the tissue, and I is the intensity of the laser beam. Units for all the variables that appear in Equation (1) are listed in Table 1. The temperature, T , or $T(x, y, z, t)$, in (1) is a three dimensional model of space, where x , y , and z (pixels) are spatial coordinates that represents the position of the tissue and t is time. The model described in Equation (1) is valid under the assumptions that the laser beam is perpendicular to the tissue surface, the tissue is homogeneous (i.e. its physical properties do not change across the tissue volume) and has a perfectly flat surface.

From the model in (1), the rate of temperature change is determined by the contribution of two terms: the laser input S and $\kappa \nabla^2 T$, that describes the temperature decrease caused by heat conduction. We can assume that S is nonzero when the laser is turned on, which will increase the temperature. We hypothesize that when the laser is on, the dynamics of tissue temperature is dominated by this term, as shown in equation (1) above. S is dependent on the intensity of the laser beam, I . Therefore, the contribution of I to the effective rate of temperature change, $\frac{1}{\rho c} S$, is mediated by the tissue density ρ , the specific heat capacity c , and the absorption coefficient α [13]. We hypothesize that if the thermal response of the tissue can be measured with a thermal camera and if that the intensity of the laser I is known, then we could potentially calculate the ρ , c , and α parameters, treating this as a nonlinear inverse problem. These parameters will then assist us in identifying the tissue type.

Symbol	Variable Name	Unit
T	Temperature of the tissue during laser exposure	$^{\circ}\text{C}$
t	Time	s
ρ	Tissue density	kg/m^3
c	Specific heat capacity	$\text{J}/\text{kg} \cdot \text{K}$
κ	Tissue temperature conductivity	$\text{W}/\text{m} \cdot ^{\circ}\text{C}$
S	Volumetric power density from the laser exposure	W/m^3
α	Coefficient of the tissue absorption	$1/\text{m}$
I	Intensity of the laser beam	W/m^2
x, y, z	Spatial coordinates representing the location of the tissue	pixels

Table 1: Table of the inputs, variable names, and units of the thermal response model (1).

While we could in principle use Equation (1) to predict the tissue type, modeling and solving this equation is a complex problem that involves nonlinearities and the application of various numerical methods. The solution of this partial differential equation depends on the form of its inputs and initial conditions, such as the laser wavelength, the type of tissue, and the range of the temperature [12]. The thermal response model described by Equation (1) depends on many laser parameters and tissue properties, therefore, solving this model and providing real-time feedback is difficult [13].

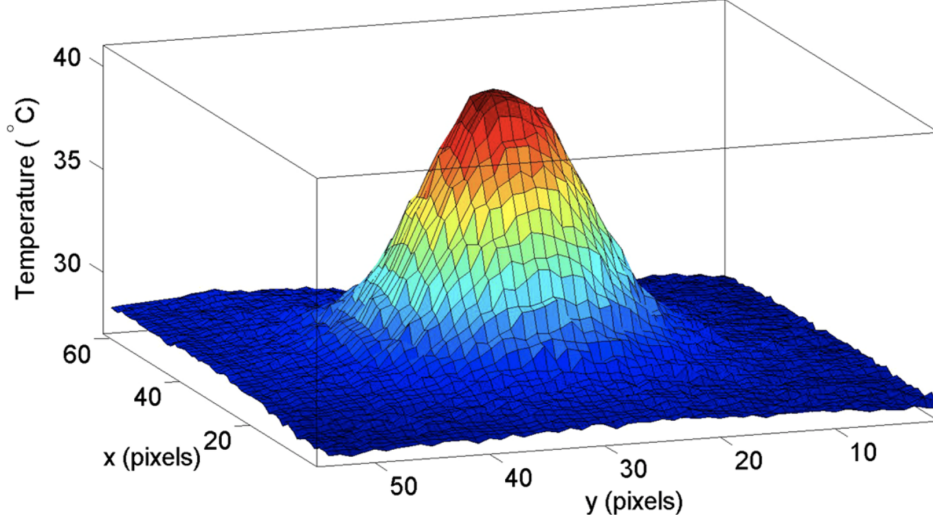


Figure 6: Temperature profile of a tissue surface captured with a thermal camera. This plot represents the temperature profile on the tissue surface 1 second after the laser exposure. The x and y axes of the plot represents the location of the tissue surface, and the z axis describes the temperature of the tissue after the laser-tissue interaction. The warm colors (including red, orange, and yellow) represents a higher temperature, and the cool colors (including green, turquoise, and blue) represents a lower temperature. Reproduced from [13].

Due to these difficulties, we want to choose an alternative temperature model that allows for quicker computation and is more straightforward to numerically compute and solve the inverse problem. As shown in Equation (1), the temperature dynamics is dependent on laser intensity, $I(W/m^2)$. When the laser is applied perpendicularly to the tissue, the laser intensity profiles of most surgical laser beams have a Gaussian profile [11]. The laser delivers the highest energy at the center and lower energy to areas further away from the center. These assumptions are corroborated by experimental findings reported in prior studies. Figure 6 displays the experimental results from Pardo et al. [13]. Based on these observations, we conclude that the Gaussian function can well represent the instantaneous temperature observed on the tissue surface.

The Gaussian model is valid under the same assumptions as thermal response model in (1). Let the tissue surface be defined as a m -by- n rectangle plane S and the time index be k which is the time (in seconds) multiplied by the frequency (in frames per second, fps). The temperature at time index k of surface S is defined as a two-dimensional matrix $T(S, k) \in \mathbb{R}_{m \times n}$ ($^{\circ}\text{C}$),

$$T(S, k) = A(k) \exp \left(- \left(\frac{(x - \mu_x(k))^2}{2\sigma_x^2(k)} + \frac{(y - \mu_y(k))^2}{2\sigma_y^2(k)} \right) \right) \quad (2)$$

where $T(S, k)$ is the temperature, $(x, y) \in S$ refers to the surface of the tissue in pixel coordinates, A is the amplitude, μ_x and μ_y represent the center location, and σ_x and σ_y are the standard deviations of the function. Further detail on the inputs of the model in (2) can be found in Table 2.

The parameters of the Gaussian model are the amplitude, A , the center location, μ_x and μ_y , and the spatial spread, σ_x and σ_y . Assuming that the position of the laser beam and the targeted tissue remains fixed during the treatment and experimental data collection, we can hypothesize that μ_x and μ_y are constant values and A , σ_x , and σ_y are changing over time. Therefore, we can define A , σ_x , and σ_y as time-varying parameters: $A(k)$, $\sigma_x(k)$ and $\sigma_y(k)$. For the entire thermal interaction process, we can collect a sequence of temperature profiles as $[T(S, k)]_{k=1,2,3,\dots}$ and can estimate the evolution of parameters, $[A(k), \sigma_x(k), \sigma_y(k)]_{k=1,2,3,\dots}$, with the estimation framework that we will introduce in Section 3.3. For different thermal responses, the temperature sequence produces different sequences of parameters.

Symbol	Variable Name	Unit
S	m-by-n rectangle plane	–
k	Time index (time multiplied by frequency)	–
$T(S, k) \in R^{m \times n}$	Tissue temperature at k of S , defined as a 2D matrix	°C
x and y	Spatial coordinates representing the location of the tissue	pixels
A	Amplitude of the Gaussian	°C
μ_x	Mean of the Gaussian in the x direction	pixels
μ_y	Mean of the Gaussian in the y direction	pixels
σ_x	Standard deviation of the Gaussian (spatial spread) in the x direction	pixels
σ_y	Standard deviation of the Gaussian (spatial spread) in the y direction	pixels

Table 2: Table of inputs, variable names, and units of the Gaussian model (2).

The thermal response of the tissue is defined by tissue properties and laser parameters. If we fix the laser settings (i.e., if we fix the volumetric power density), the evolution of the parameters could provide an approximation of the tissue properties. Therefore, we advance the hypothesis that we can distinguish between different types of tissue by comparing the evolution of the parameters over time.

3.2 Computer Simulation of Sequential Temperature Data

To verify the hypothesis described in Section 3.1, i.e., that we can determine the tissue type by studying the evolution of parameters, we decided to use the Gaussian model in Equation (2) to generate a synthetic data set of thermal laser-tissue interactions. We chose to model the temperature profile at a single time step with the Gaussian model, and defined the amplitude, $A(k)$, and the standard deviation, $\sigma_x(k)$ and $\sigma_y(k)$, as time-varying variables. In our data simulations, we assume that the tissue has even texture and properties over the surface, so we can simplify the original Gaussian model by defining equal standard deviations in x and y axes, which means $\sigma_x = \sigma_y = \sigma$. We want to develop a computer simulation code that consumes time-varying functions of amplitude A and standard deviation σ based on k , and produces realistic series of temperature data with visualizations. To achieve this goal, we need to define appropriate parameter functions for $A(k)$ and $\sigma(k)$ that can present the change in tissue temperature during a laser exposure.

To better simulate the behavior of the parameters A and σ during the laser exposure, we referred to the results of the tissue-laser experiment described in Pardo et al. [13]. As shown in Figure 7, the surface temperature undergoes two phases during a laser-tissue interaction: heating (when the laser is turned on) and cooling (when the laser is turned off). During the heating phase, A and σ behaves differently from the cooling phase. Therefore, we can define piecewise functions, in terms of time t , to model the change in the A and σ parameters during the entire laser-tissue interaction.

To mimic the laser-tissue interaction shown in Figure 7, we need to study the behaviors of A and σ during heating and cooling. As Figure 7 starts at the heating phase (at $t = 5.8$ s, instead of $t = 0$ s), we want to assume that the initial amplitude is at 0°C and that the initial standard deviation value is 0 pixels. We also assume that the initial amplitude is in the absence of laser application, such that at $t = 0$ s, the tissue temperature is at 0°C after subtracting from the baseline temperature. We hypothesize that the amplitude will have a rapid increase from 0°C, followed by a more mild increase to 80°C, then to a logarithmic decay when it cools down.

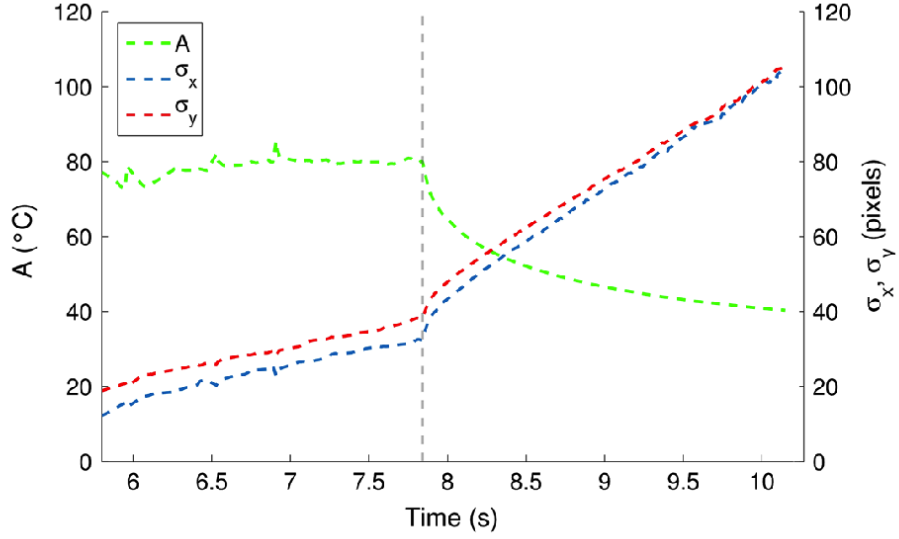


Figure 7: The variation of $(A, \sigma_x^2, \sigma_y^2)$ during laser-tissue interaction. The section to the left of the vertical line on the graph displays the heating phase and the section on the right displays the cooling phase of the tissue after laser exposure. Reproduced from [13].

We defined the behavior of A and sigma as follows:

$$A(t) = \begin{cases} 140t, & t < 0.5 \\ 65 + 10t, & t < 2 \\ 85 - 40 \log(t - 1), & t > 2 \end{cases} \quad (3)$$

$$\sigma(t) = \begin{cases} 15t, & t < 2 \\ -10 + 20t, & t > 2 \end{cases} \quad (4)$$

The plots of these functions is displayed in Figure 8. We simulate sequential temperature data from $t = 0$ s to 5 s, where the heating phase is from $t = 0$ s to 2 s and the cooling phase is from $t = 2$ s to 5 s. In the heating phase, we describe the amplitude using a linear function with a slope of 140 and a y-intercept of 0 to simulate a short but rapid increase in the first 0.5 seconds and another linear function with a slope of 10 and a y-intercept of 65 to simulate a slower increase in amplitude. In the cooling phase, we generate the amplitude with a logarithmic decay to describe the declining amplitude. For the standard deviation σ , we use an increasing linear function with a slope of 15 to describe the increasing spatial spread during the heating phase and another linear function with a higher slope of 20 and a y-intercept of -10 to represent the faster increment in the cooling phase. Figure 8 shows a plot of these parameters over time.

To mimic real-world data as closely as possible, we add Gaussian noise to our simulated temperature profiles. Additive noise can be caused by the hardware of the thermal camera and uneven tissue properties, as well as noises generated during the processing of raw data from the thermal camera. Figure 9 displays an example of a simulated temperature surface at $t = 2$ s. In the next section, we introduce the estimation framework that is verified with the simulation data.

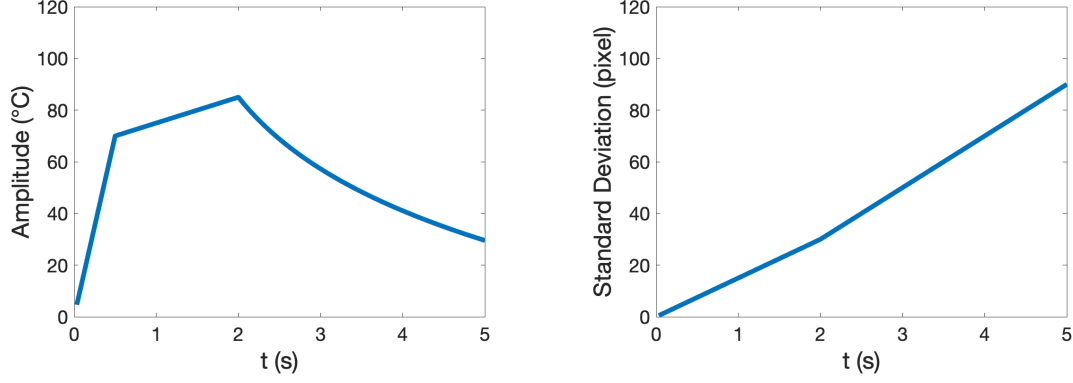


Figure 8: Evolution of A (left) and σ (right) with linear heating and logarithmic cooling. The A and σ parameter functions share the characteristics of the parameter evolution graph from Pardo et al. [13], which is shown in Figure 7.

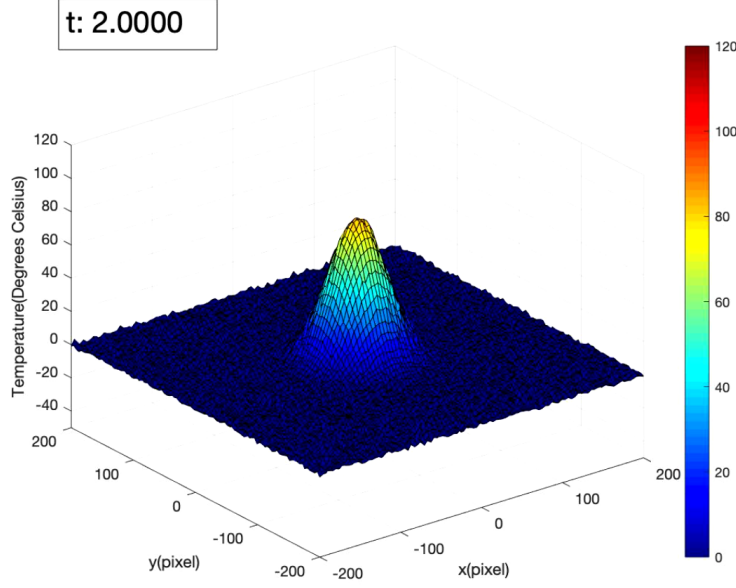


Figure 9: The plot of simulated temperature data at $t = 2s$, with amplitude defined in (3) and standard deviation defined in (4), and at each location on the surface there is a random noise ($^{\circ}\text{C}$) added to the temperature. The random noise ($^{\circ}\text{C}$) is from a normal distribution that centered at 0 ($^{\circ}\text{C}$) and has a standard deviation of 1 (pixel).

3.3 State and Parameter Estimation Framework

As discussed in Section 3.1, we hypothesize that different types of tissue vary in their capacities to absorb and diffuse heat during thermal interaction and have unique thermal responses to the laser. Since the thermal responses are defined by the behaviors of the time-varying parameters A and σ , if our hypothesis is correct, it is possible to distinguish between different types of tissue, based on the estimations of the time-varying parameters. Therefore, with the observed temperature data we want to estimate the sequence of amplitude A and standard deviation σ . To accomplish this, we utilize the Bayesian nonlinear filtering framework, specifically the Ensemble Kalman Filter (EnKF).

In this section, we illustrate the application of the EnKF estimation framework through an example of a set of simulated temperature variations. More details on the EnKF can be found in [14] and references therein. The input temperature sequence, which is also called observation data here, is a simulation of thermal interaction on a 100-by-100 flat tissue surface. The duration of the simulation is 10 seconds, and the frequency is 30 frames per second (fps). In this simulation, we assume the laser beam is perpendicular to the tissue, and therefore, the temperature profile is symmetric. We also have random noises added to each temperature matrix in the sequence to simulate the errors of the thermal camera in a real data collection.

The observation data is a discrete 3-dimension array, $[T_1, T_2, \dots, T_j, \dots, T_{300}] \in \mathbb{R}_{300 \times 100 \times 100}$, where each $T_j \in \mathbb{R}_{100 \times 100}$ and j represents the discrete time index at time t ,

$$j = frequency \times t, \quad 0 < t \leq 10, \quad 0 < j \leq 300 \quad (5)$$

3.3.1 Overview of the Nonlinear Filtering

With the Gaussian model in (2), we have the unknown model state T and unknown model parameters A and σ . We treat these unknowns as random variables with probability distributions. By Bayes's Theorem, the posterior density, which is the density of A , σ and T given the evidence T_{obs} , is:

$$\pi(A, \sigma, T | T_{obs}) \propto \pi(T_{obs} | A, \sigma, T) \pi(A, \sigma, T) \quad (6)$$

In (6), $\pi(T_{obs} | A, \sigma, T)$ is the likelihood function that indicates if the model parameters and state are given, how likely can we observe the temperature T_{obs} . The prior distribution $\pi(A, \sigma, T)$ is the probability distribution of the parameters jointly with the states. We let the accumulated observation data from time index 1 to j to be D_j ,

$$D_j = \{T_{obs,1}, T_{obs,2}, \dots, T_{obs,j}\} \quad (7)$$

and we are interested in the posterior density $\pi(A_j, \sigma_j, T_j | D_j)$, which describes the density of model states and parameters given the cumulative observation data D_j . We want to consecutively update the posterior density with a ‘‘predictor-corrector-type scheme’’ [14]:

$$\pi(A_j, \sigma_j, T_j | D_j) \longrightarrow \pi(A_{j+1}, \sigma_{j+1}, T_{j+1} | D_j) \longrightarrow \pi(A_{j+1}, \sigma_{j+1}, T_{j+1} | D_{j+1}) \quad (8)$$

At each iteration, the first step is to predict the model state T_{j+1} with the parameters at j and predict the parameters A_{j+1} and σ_{j+1} by adding drift terms to A_j and σ_j . The second step is to correct the prediction we made by the temperature profile at index $j+1$, which is T_{j+1} . In this example, we need to repeat this scheme from time index 1 to 300 and we can get the posterior density at each time index.

3.3.2 Pre-Process the Discrete Temperature Sequence

Before we can apply nonlinear filtering to the temperature sequence, we need to parse it and generate a smaller data set that can still represents the temperature evolution, because the original temperature sequence is large in size, which results in a longer execution time of the estimation. Our solution is to simplify the data by selecting points from the original temperature sequence in a grid pattern (e.g. selecting points from a 100×100 grid to a 20×20 grid). In this example, we process the simulated temperature sequence to a 20-by-20 grid. The processed temperature sequence is represented as $T_{obs}^p = [T_{obs,1}^p, T_{obs,2}^p, T_{obs,3}^p, \dots, T_{obs,j}^p, \dots, T_{obs,300}^p]$, and each temperature matrix in the sequence is processed to the following $T_{obs,j}^p$ matrix:

$$T_{obs,j}^p = \begin{bmatrix} T_{obs,j(1,1)} & T_{obs,j(1,6)} & \cdots & T_{obs,j(1,96)} \\ T_{obs,j(6,1)} & T_{obs,j(6,6)} & \cdots & T_{obs,j(6,96)} \\ \vdots & \vdots & \ddots & \vdots \\ T_{obs,j(96,1)} & T_{obs,j(96,6)} & \cdots & T_{obs,j(96,96)} \end{bmatrix} \in \mathbb{R}_{20 \times 20}, \quad j = 1, 2, 3, \dots, 300 \quad (9)$$

Notice that in the following sections, for easier reading, we will use T_{obs} to reference the parsed observed temperature data.

3.3.3 Initialization of the Ensemble

After parsing the observed data, we need to construct the initial ensemble that consists of random draws from the initial prior distribution. The ensemble size in this example is defined as N_{ens} and the ensemble at time index j is represented by $S_{j|j}$ and has the following form:

$$S_{j|j} = \left\{ (T_{j|j}^1, A_{j|j}^1, \sigma_{j|j}^1), (T_{j|j}^2, A_{j|j}^2, \sigma_{j|j}^2), \dots, (T_{j|j}^{N_{ens}}, A_{j|j}^{N_{ens}}, \sigma_{j|j}^{N_{ens}}) \right\} \quad (10)$$

For storage purpose, we also need to initialize the prior distribution of temperature states as $T_{0|0}^n$. We assume that after we subtract the baseline temperature from the temperature at each spatial point, the tissue prior temperature is constantly 0°C everywhere on its surface:

$$T_{0|0}^n = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix} \in \mathbb{R}_{20 \times 20}, \quad n = 1, 2, 3, \dots, N_{ens} \quad (11)$$

Here we choose to initialize the prior distributions of $A_{0|0}^n$ and $\sigma_{0|0}^n$ with Uniform distributions because they are symmetric and random data points within the ranges are equally likely to be drawn. In the Results chapter we will provide more details on the effects of choosing different types of distributions in initializing the ensemble.

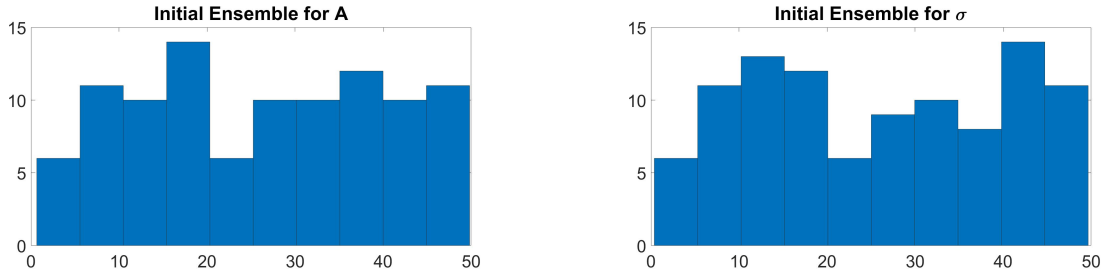


Figure 10: The histograms of random draws from the prior distributions. For each parameter, A (left) and σ (right) we drew 100 data points from the Uniform distribution defined by $Uniform([0, 50])$.

3.3.4 Prediction Step

As described in Section 3.3.1, the EnKF framework consists of a prediction step and an observation update step. In the prediction step, as we move from time index j to $j+1$, the first step is to predict the temperature states and parameters at $j+1$. The temperature states are predicted with the corrected parameters at index j and the Gaussian model (2):

$$T_{j+1|j}^n = Gauss(A_{j|j}^n, \sigma_{j|j}^n) = A_{j|j}^n \cdot \exp \left(- \left(\frac{x^2}{2(\sigma_{j|j}^n)^2} + \frac{y^2}{2(\sigma_{j|j}^n)^2} \right) \right) + C \in \mathbb{R}_{20 \times 20} \quad (12)$$

where $C \in \mathbb{R}_{20 \times 20}$ represents the modeling error. In terms of the parameters, since the parameters are changing on a slow scale, we model the evolution of parameters as a random walk and predict the parameters by adding drift terms to them [15]:

$$A_{j+1|j}^n = A_{j|j}^n + E_A^n \quad (13)$$

$$\sigma_{j+1|j}^n = \sigma_{j|j}^n + E_\sigma^n \quad (14)$$

where E_A^n and $E_\sigma^n \in \mathbb{R}$ are random draws generated from a Normal distribution with a mean of 0 and a specific standard deviation. Then, the temperature matrix is reshaped to a single column vector y :

$$y_{j+1|j}^n = \text{reshape}(T_{j+1|j}^n) = \begin{bmatrix} T_{j+1|j(1,1)}^n \\ T_{j+1|j(2,1)}^n \\ \vdots \\ T_{j+1|j(20,1)}^n \\ T_{j+1|j(1,2)}^n \\ T_{j+1|j(2,2)}^n \\ \vdots \\ T_{j+1|j(20,2)}^n \\ \vdots \\ T_{j+1|j(20,20)}^n \end{bmatrix} \in \mathbb{R}_{400 \times 1} \quad (15)$$

The prediction ensemble statistics $z_{j+1|j}^n$ are defined by the reshaped predicted temperature states $y_{j+1|j}^n$ and parameter values at time $j+1$ as:

$$z_{j+1|j}^n = \begin{bmatrix} y_{j+1|j}^n \\ A_{j+1|j}^n \\ \sigma_{j+1|j}^n \end{bmatrix} \in \mathbb{R}_{402 \times 1}, \quad n = 1, 2, 3, \dots, N_{ens} \quad (16)$$

3.3.5 Observation Update Step

With the prediction of $z_{j+1|j}^n$, in the observation update step, we correct our prediction by taking the observed temperature data $T_{obs,j+1}$ into account. Specifically, we want to calculate the Kalman gain, K_{j+1} , which reflects how amplitude A and standard deviation σ correlate to the tissue temperature (in other words, the correlation between the states and parameters of the model), and use the Kalman gain to update the prediction. To achieve this, we first need to reshape the matrix $T_{obs,j+1}$ to get a single column observed data vector $y_{obs,j+1}$:

$$y_{obs,j+1} = \text{reshape}(T_{obs,j+1}) = \begin{bmatrix} T_{obs,j+1(1,1)} \\ T_{obs,j+1(2,1)} \\ \vdots \\ T_{obs,j+1(20,1)} \\ T_{obs,j+1(1,2)} \\ T_{obs,j+1(2,2)} \\ \vdots \\ T_{obs,j+1(20,2)} \\ \vdots \\ T_{obs,j+1(20,20)} \end{bmatrix} \in \mathbb{R}_{400 \times 1} \quad (17)$$

The Kalman gain is computed as:

$$K_{j+1} = \Sigma_{j+1}^{zy} \left(\Sigma_{j+1}^{yy} + D_{j+1} \right)^{-1} \quad (18)$$

in which Σ_{j+1}^{zy} represents the cross covariance of the augmented predictions $z_{j+1|j}^n$ and the observation prediction $y_{j+1|j}^n$, Σ_{j+1}^{yy} represents the forecast error covariance of the observation prediction ensemble $y_{j+1|j}^n$,

and D_{j+1} represents the observation error covariance [14]. The cross covariance matrix Σ_{j+1}^{zy} is computed by:

$$\Sigma_{j+1}^{zy} = \frac{1}{N_{ens} - 1} \cdot \sum_{j=1}^{N_{ens}} \left((z_{j+1|j}^n - \hat{z}_{j+1|j}) (y_{j+1|j}^n - \hat{y}_{j+1|j})^\top \right) \quad (19)$$

in which the prediction ensemble mean is computed using the following formula:

$$\hat{z}_{j+1|j} = \frac{1}{N_{ens}} \cdot \sum_{n=1}^{N_{ens}} z_{j+1|j}^n \quad (20)$$

The forecast error covariance Σ_{j+1}^{yy} is computed by:

$$\Sigma_{j+1}^{yy} = \frac{1}{N_{ens} - 1} \cdot \sum_{j=1}^{N_{ens}} \left((y_{j+1|j}^n - \hat{y}_{j+1|j}) (y_{j+1|j}^n - \hat{y}_{j+1|j})^\top \right) \quad (21)$$

in which the mean of the reshaped predicted temperature states $y_{j+1|j}^n$ is defined as:

$$\hat{y}_{j+1|j} = \frac{1}{N_{ens}} \cdot \sum_{n=1}^{N_{ens}} y_{j+1|j}^n \quad (22)$$

Afterwards, we can use the Kalman gain and the observed data at time index $t + 1$ to correct the augmented posterior ensemble $z_{j+1|j+1}^n$, which is written as:

$$z_{j+1|j+1}^n = z_{j+1|j}^n + K_{j+1} (y_{obs,j+1} - y_{j+1|j}^n) = \begin{bmatrix} y_{j+1|j+1}^n \\ A_{j+1|j+1}^n \\ \sigma_{j+1|j+1}^n \end{bmatrix} \in \mathbb{R}_{402 \times 1}, \quad n = 1, 2, 3, \dots, N_{ens} \quad (23)$$

where $y_{obs,j+1}$ is the observed data and $y_{j+1|j}^n$ is the reshaped predicted temperature state, whose definition is shown in Equation (15). With the augmented posterior ensemble $z_{j+1|j+1}^n$, we can compute the model parameter estimates at time index $j + 1$, which are given by the posterior ensemble means :

$$\hat{A}_{j+1|j+1} = \frac{1}{N_{ens}} \cdot \sum_{n=1}^{N_{ens}} A_{j+1|j+1}^n \quad (24)$$

$$\hat{\sigma}_{j+1|j+1} = \frac{1}{N_{ens}} \cdot \sum_{n=1}^{N_{ens}} \sigma_{j+1|j+1}^n \quad (25)$$

We repeat this prediction-correction procedure for each j from 1 to the end of the time indexing, which is 300 in this example.

3.3.6 Outputs of the Estimation Framework

By repeating the prediction-correction process on the temperature sequence from time index 1 to 300, we can get the estimation of time-varying parameters A and σ as:

$$A = [\hat{A}_{1|1}, \hat{A}_{2|2}, \dots, \hat{A}_{300|300}] \quad (26)$$

$$\sigma = [\hat{\sigma}_{1|1}, \hat{\sigma}_{2|2}, \dots, \hat{\sigma}_{300|300}] \quad (27)$$

As described in Section 3.1, we hypothesize that various types of tissue have unique thermal responses and we can classify them based on these thermal responses. Since the time-varying parameters A and σ are the characteristics that define the thermal responses, the uniqueness would be reflected in the unique behaviors of A and σ . Therefore, it opens up the possibilities to classify different types of tissue based on the estimation of the parameters. We will introduce how the classifier is designed and developed in the next section.

3.4 Machine Learning Models and Techniques for Classification of Thermal Data

Having described the methods to estimate the temperature amplitude and spread over time, we now focus on the problem of using this information to infer the underlying tissue state. We formulate this problem in the form of a supervised classification problem.

3.4.1 Implementation of the Classifier

From the estimation framework described in the previous section, we obtain the amplitude A and standard deviation σ over each time step. Although it might be plausible to use these estimation sequences to directly classify tissue types, it would be impractical due to the dimensionality of the data set. Instead, we opted to use derived features of the estimated A and σ sequences. The four features that were studied are the mean and maximum of both estimated sequences. Classification methods that were utilized in this study are Linear Discriminant Analysis (LDA) and Support Vector Machines (SVM). Implementations of these classifications models were created using MATLAB 2019b with the help of the Classification Learner application.

To allow for classification, we must identify features that allow for distinguishing of data types. From the estimation model as described in Section 3.3, we are presented with three sequences over time: A , σ_x , and σ_y , representing the amplitude, standard deviation in the x and y directions, respectively. For the purpose of conducting exploratory data analysis and understand what features to use, we generated a synthetic dataset of thermal interactions using the simulation software described in the previous section. We simulated the interaction of a laser with two types of tissue, which we shall identify as “Type A” and “Type B”. Type B is a tissue with more pronounced absorption, and it can reach temperatures of up to 220°C during an interaction. By contrast, Type A can only reach up to 170°C. The plots in Figure 11 show the features extracted from these interactions. Each data point is representative of an 8-second interaction, captured at 30 Hz, meaning that the total simulation consisted of 240 image frames. From these sequences of frames, we extract the four chosen classification features identified above. Exploratory data analysis suggests that the max amplitude, mean amplitude and max standard deviation enable to clearly separate data points representative of different tissue states (see the left plot in Figure 11). Ineffective derived features for classification were also studied, as shown in the right plot in Figure 11, where there is no clear separation between the clusters of data along the axes, and would not allow for effective classification.

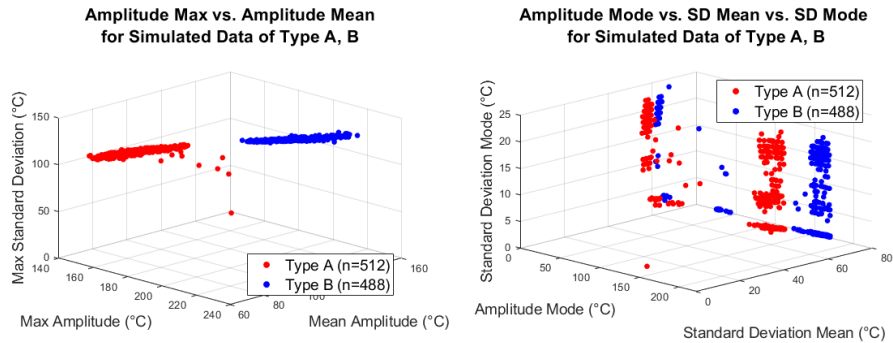


Figure 11: These two plots identify the classification different features spaces of possible derived parameters of the estimation framework sequences. Here 6 different features are shown, the mean, mode and maximum of both the estimated amplitude (A) and standard deviation (σ). The red dots represent simulated data points of Type A, generated with a maximum amplitude of 170°C. The blue dots, simulated data of Type B with a maximum amplitude of 220°C. Effective classification features are those that place the red and blue dots in distinguishable clusters of each type.

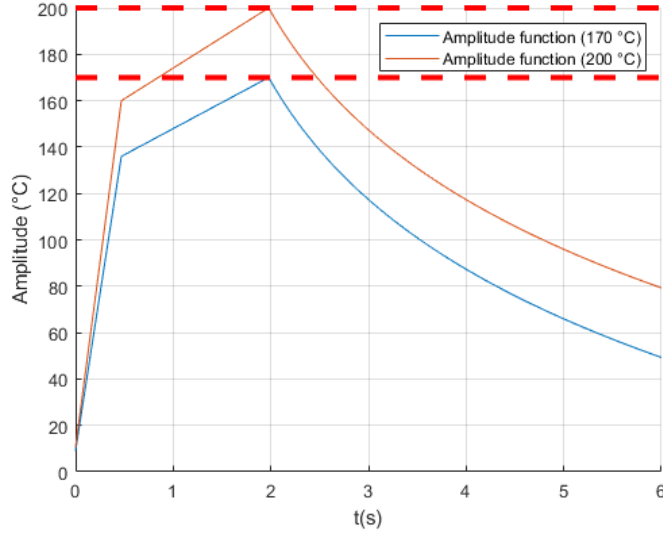


Figure 12: An example of two different amplitude functions used to generate tissue types. One amplitude function has a maximum amplitude, A_{max} , of 170°C and the other 200°C. These tissue types are then labeled as Type A or Type B, respectively.

3.4.2 Simulated Data Generation

From these four features, we generated simulated data to train different classification models and test their robustness. Each classification model is tested on N training data points, where $N = 1000$. Each training data point is a five dimensional vector that contains, a , a vector of four selected features and, b , a tissue type training label. The vector of features, a , consists of four values: The maximum value of the amplitude sequence, A_{max} ; the mean of the amplitude sequence, A_{mean} ; the maximum of the standard deviation sequence, σ_{max} ; and the mean of the standard deviation sequence, σ_{mean} . All these features are calculated from the output of the estimation framework described in Section 3.3.6. The variable b represents the true label of the simulated data, a 0 or 1, and that the classification model must correctly predict. A value of 0 represents an assigned Type A and 1 assigned Type B.

$$a = [A_{max}, A_{mean}, \sigma_{max}, \sigma_{mean}], \quad b = 0, 1 \quad (28)$$

To generate training data on a wide range of thermal data spreads, the parameters listed in Table 3 were modified. Changing the simulation parameters affected the classification results for the classification models. Amplitude maximum A_{max} , represents the maximum temperature that the Gaussian simulation approaches over time. The amplitude noise is an absolute value that increases or decreases the maximum temperature of the simulation. The observation noise changes the value of every pixel in the simulation over time. Both the observation and amplitude noise are an effort to account for the noise found in experimental data obtained from a thermal camera.

Parameter	Description	Unit
Amplitude Maximum (A_{max})	The peak temperature of the Gaussian simulation.	°C
Amplitude Maximum Noise	The absolute noise on the maximum amplitude function of the simulated data.	°C

Table 3: Parameters to modify simulated data. These parameters were modified to train different classification models and study the accuracy.

Parameter	Description	Default Value
Simulation Length	The length in seconds of the simulated data	8 seconds
Frequency	The number of frames per second	30 Hz
Amplitude Lower	Lower endpoint of the amplitude in the Uniform distribution	5
Amplitude Upper	Upper limit of the amplitude in the Uniform distribution	30
Ensemble Sample Size (N_{ens})	Sample size of the ensemble distribution	500
Observation Noise (D)	Degree to which the estimated framework trusts data points	0.01
Model Prediction Error (C)	Degree to which the estimated parameters are trusted	1e-10
Grid Size	Size of the sampled grid	10

Table 4: Parameters of the estimation framework to generate simulated data.

Experiment Set	Title	Description	Tested Values
1	Amplitude Difference	Test the maximum amplitude difference between two generated types that the amplitude is able to distinguish	50, 30, 10, 5°C
2	Training Robustness	Test the a model that was trained on types of data with a specific amplitude difference on a higher and lower amplitude difference	50, 30, 10, 5
3	Amplitude Noise	Test the effect of noise on the maximum amplitude on classification	10, 20°C

Table 5: Different experiment sets that were executed to test the SVM and LDA classification models.

The simulated data sequence was passed to the Estimated Kalman Filter to obtain sequences of estimated A and σ over time from which the features of the classification model are calculated. Table 4 denotes the used settings of the estimation framework.

3.4.3 Classification Testing

After the training of the classification models was completed, we tested the accuracy of the models with newly generated data points. From this, we obtained the accuracy of the classification along with a confusion matrix displaying the predictions for each generated type of data. Table 5, provides a summary of executed experiments. Experiments were performed on both the SVM and LDA classification models and results detailed in Section 4.4.

The testing data was created in the same manner as described in Section 3.4.2. However, the total number of testing data points, N , is 100. This value is reduced from 1000, which was used in training the models due to the significant amount of time needed for the generation of data. Generating 100 data samples took about 90 minutes. For each sample, a tissue was generated as either Type A or Type B according to a uniform random distribution. The assigned label, b , was used to verify the prediction of the model after classification. Results of the classification experiments can be found in Section 4.4.

4 Numerical Results

4.1 Data Simulation and Parameter Estimation: Single Pulse

Here we present the parameter estimation results using simulated data of thermal laser-tissue interactions for a single pulse. As described in Section 3.2, the evolution of the Gaussian parameters can be modeled with piecewise functions, presenting the heating and cooling phases of the tissue during laser exposure. Assuming that $\mu_x = \mu_y = \mu = 0$, we can simulate a temperature data set for a single pulse. For each time step, we can compute the amplitude A and standard deviation σ , using the $A(t)$ and $\sigma(t)$ piecewise functions, defined by Equations (3) and (4). Then, we can calculate the tissue temperature by plugging the sets of computed $A(t)$ and $\sigma(t)$ into the Gaussian model (2) with additive noise. This results in the tissue temperature at each time step, or the simulated temperature data set. Figure 13 provides the visualization of a generated temperature data set and Figure 14 displays the parameter functions used to simulate a set of temperature data for a single pulse.

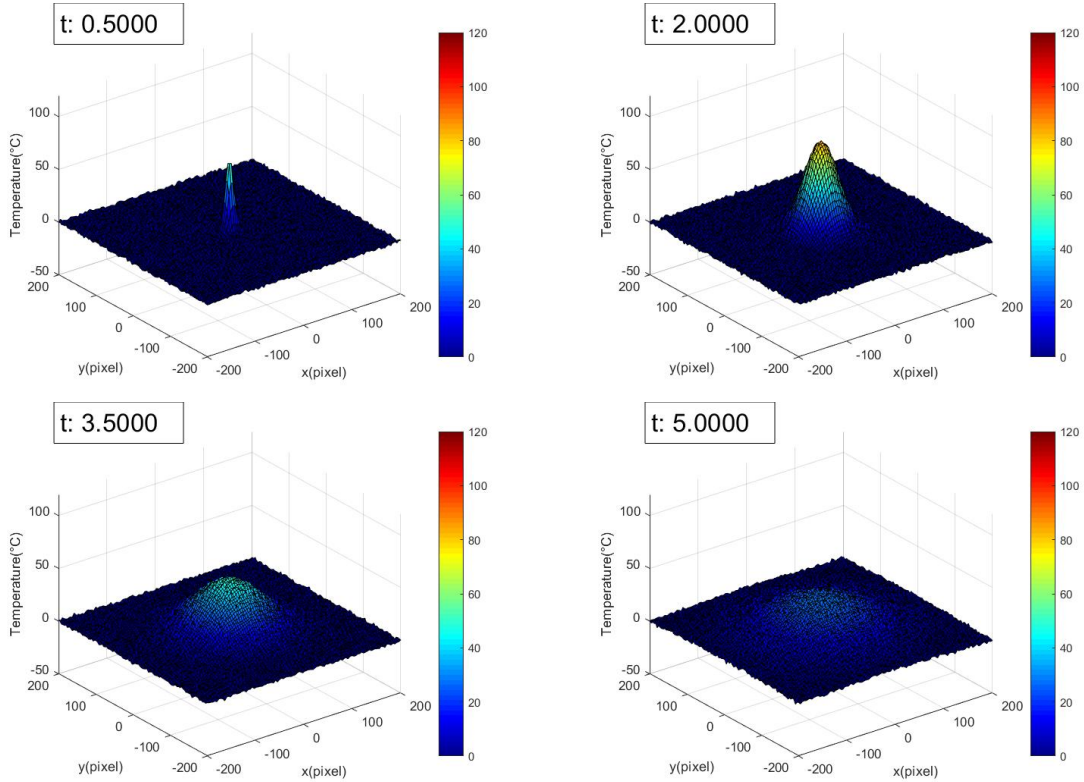


Figure 13: Visualizations of the simulation for a single pulse, displaying the behaviors of the amplitude and standard deviation parameters at $t = 0.5$ s (top left), $t = 2$ s (top right), $t = 3.5$ s (bottom left), $t = 5$ s (bottom right). This data set is simulated with the parameter functions displayed in Figure 14.

With the simulated temperature data set, we employ the EnKF framework described in Section 3.3 and study the accuracy of the resulting parameter estimations. Figure 15 displays the resulting parameter estimations using the data set generated from the parameters shown in Figure 14. To quantitatively measure the accuracy of the parameter estimations, we computed the absolute error of the estimations, by finding the absolute difference of the the true model parameters and estimated model parameters over time. Figure 16 displays the absolute errors between the true model parameters and the EnKF mean estimates over time. For $t < 1$ s, the estimated A and σ are less accurate, as the absolute errors of the estimated A and σ oscillates between 0 and 25. We hypothesize that this is caused by the time needed for the framework to capture the

change in A and σ . As shown in Figure 16, the estimated parameters are close to the model parameters after $t = 1$ s, as the absolute errors of A and σ are minimal and close to zero.

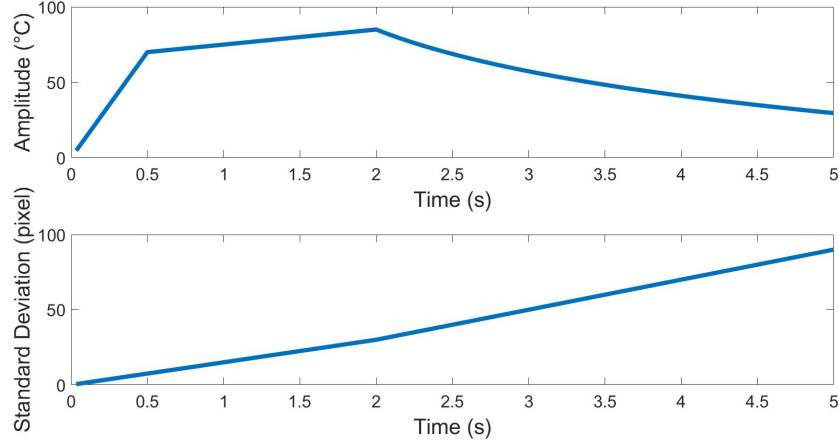


Figure 14: Amplitude and standard deviation parameter functions used to simulate a set of temperature data for a single pulse displayed in Figure 13.

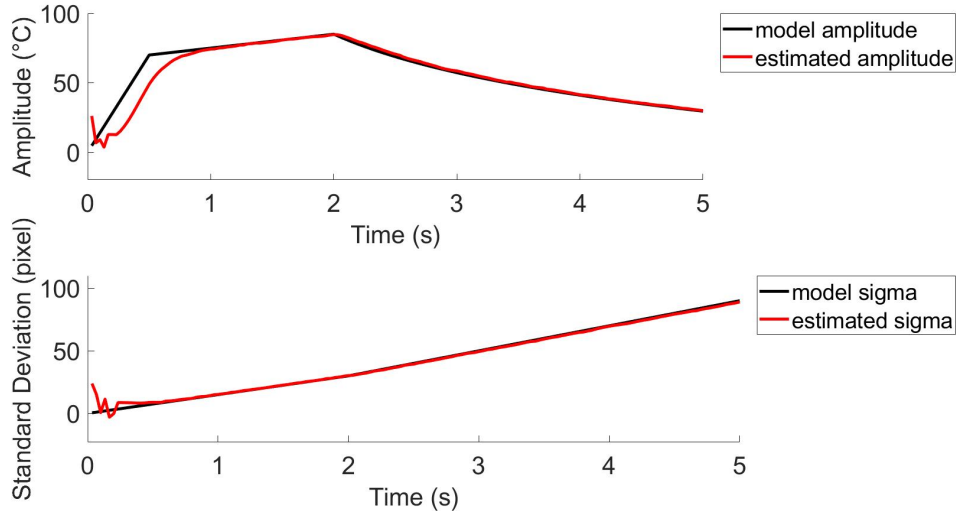


Figure 15: Parameter estimations for a single pulse. These plots display the model parameters (black) and estimated parameter or the EnKF mean estimates (red).

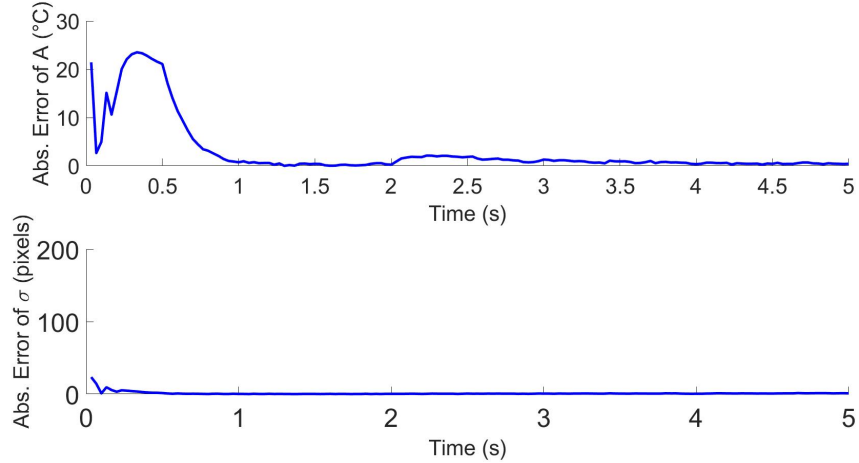


Figure 16: Absolute errors of the A and σ estimations, shown in Figure 15, over time. The absolute error is calculated by computing the absolute difference of the the true model parameters and estimated model parameters over time.

4.2 Data Simulation and Parameter Estimation: Multiple Pulses

4.2.1 Additional Pulse is Delivered After the Cooling of Previous Pulse

Here we present the results for generates sets of simulated data of thermal laser-tissue interactions for multiple pulses. After the verification of the framework with the simulation of a single pulse laser-tissue interaction, we decided to simulate data for the case of multiple pulses and verify that our estimation framework can work for multi-pulse temperature data. Multi-pulse temperature simulation and estimation are vital because in actual laser treatments, the surgeon usually require multiple laser pulse deliveries to treat the harmful tissue. In the case of multiple pulses, the framework should be able to give accurate estimations of the time-varying parameters to provide accurate feedback to the surgeon.

To simulate data for multiple pulses, we start with the assumption that the next pulse will always start after the tissue temperature cools down to the base temperature. We assume that the base temperature of the tissue after subtracting from the baseline temperature, is 0 °C. Additionally, we defined a set time for each pulse, such that the tissue heats and cools within the same given time for every pulse. Therefore, the total time range of the simulation is extended to the corresponding number of laser pulses. For example, if laser-tissue interaction takes 10 s for a single laser pulse, then the time range for a two-pulse simulation should be 20 s.

With the assumption that each pulse is delivered after the tissue cools down to the base temperature, we built upon the data generation process to account for the case of multiple pulses. We define the time-varying parameters, A and σ , as periodic and each period is a single pulse. The behaviors of the parameters remain the same from pulse to pulse. In general, we are repeating the parameter functions for a single pulse to account for the additional pulses. Referring to the previous example of a two-pulse simulation, we repeated the single pulse simulation twice, for every 10 s until the extended total time range of 20 s is reached. Figure 17 displays the simulated parameter functions of a two-pulse laser-tissue interaction.

Once the data simulation incorporates the repeated pulses, we tested and verified the parameter estimation framework with the temperature data set simulated from the parameter functions shown in Figure 17. The estimation results from the estimation framework are displayed in Figure 18. The absolute errors between the true model parameters and the EnKF mean estimates over time are displayed in Figure 19. According to this figure, the absolute error is high at the start of each pulse (at $t = 0$ s to 1 s and $t = 10$ s

to 12 s), which indicates that the estimations are less accurate. Over the time of each pulse (at $t = 1$ s to 10 s and $t = 12$ s to 20 s), the estimations become more close to the model parameters as the absolute error becomes smaller and close to 0. For each pulse, the estimation framework take about one to two seconds until the results become reliable. We hypothesize that this is because the estimation frameworks needs time to capture the change in amplitude and standard deviation of the next pulse.

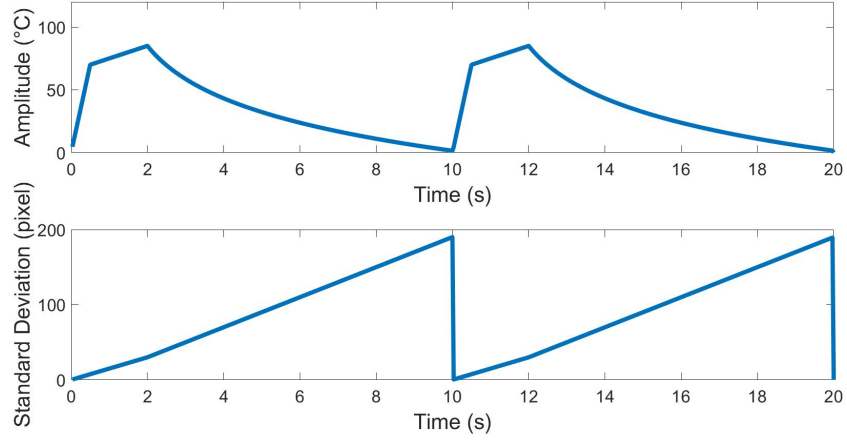


Figure 17: Amplitude and standard deviation functions used to generate a temperature data set for two laser pulses. In this case, the second pulse is delivered after the tissue cools down to its base temperature of 0 °C.

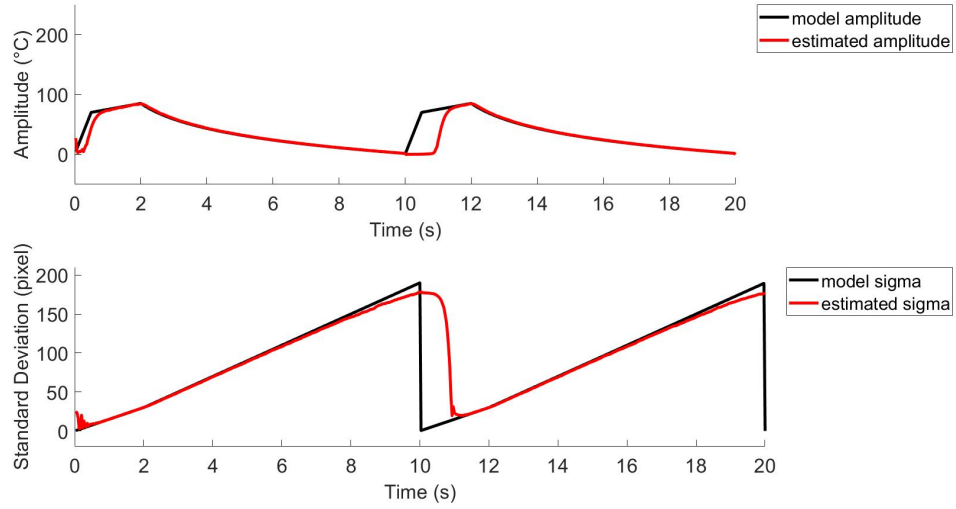


Figure 18: Estimation results of the EnKF framework for the two pulse simulation displayed in Figure 17. The true model parameters are shown in black and the EnKF mean estimates are shown in red.

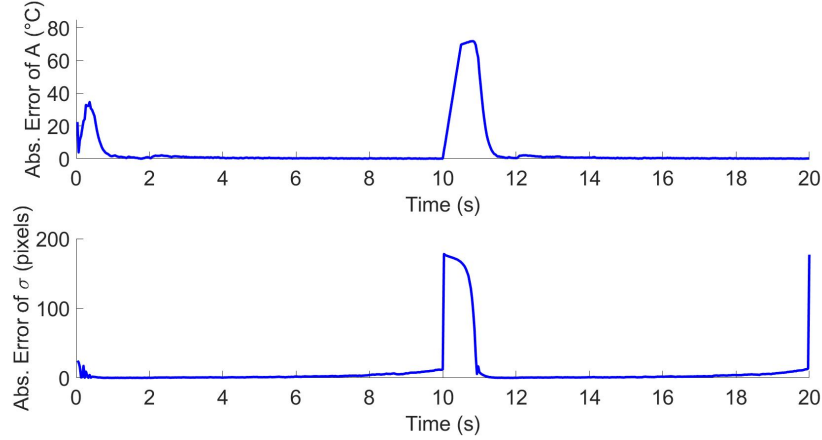


Figure 19: Absolute errors of the A and σ estimations, shown in Figure 18, over time. The absolute error is calculated by computing the absolute difference of the the true model parameters and estimated model parameters over time.

4.2.2 Residual Heat

In the previous section, we have verified the EnKF framework with multiple pulses temperature data, where each laser pulse is applied after the tissue completely cooled down from the previous laser pulse to the base temperature. However, in real-world laser-tissue treatments, the surgeon may apply the next laser pulse when the tissue is only partially cooled down. Due to the overlapping of two pulses, during the heating phase of the next pulse, there should exist a residual heat from the previous pulse. As residual heat is the heat that remains after the tissue partially cools down, we hypothesize that the existence of residual heat is reflected in the behaviors of the time-varying parameters. The amplitude is expected to increase from pulse to pulse. The spatial spread is expected to restart at the base spread of 0 pixels and be increasing throughout every additional pulse.

With these hypotheses, we need to revise the piecewise $A(t)$ and $\sigma(t)$ functions of the simulation. The new parameter functions for residual heat, $A(t, l, p)$ and $\sigma(t, l, p)$, will now take in the time index t , the total duration of each pulse l , and the pulse number p . The new amplitude function is still periodic with the addition of residual to each period, or pulse. $A(t, l, p)$ involves a repeated transformation of the original $A(t)$ function. Essentially, $A(t)$ is repeatedly transformed by the endpoint of the previous pulse, which is the tissue temperature $A(l)$ at time l , or $(l, A(l))$. We are transforming the $A(t)$ function l s to the right and $A(l)$ °C up.

We hypothesize that even with the addition of pulses, the spatial spread should not be increase continuously over time, but restart at the base spread of 0 pixels. When a laser pulse is delivered to the tissue, the heat is less spread out over the surface, resulting in a small standard deviation. As the tissue cools down, the standard deviation increases. Applying this assumption to the new σ function, $\sigma(t, l, p)$, the behavior of the σ parameter for a single-pulse should repeat from pulse to pulse. For every additional pulse, the spatial spread will first decrease significantly and then increase over time. Figure 20 displays plots that reflect the behavior of the amplitude and standard deviation of a multiple pulse simulation with the addition of residual heat.

The additional input parameters (l and p) of the $A(t, l, p)$ and $\sigma(t, l, p)$ functions allow more flexibility in the data generation process. It can generate simulated data with different pulse duration (l) and numbers of pulses (p) more efficiently, by simply redefining the input variables. Figure 21 shows the plots of the parameter functions used to generate a different simulation from Figure 20, by simply redefining the value of l from 5 to 6.

With a simulated set of data for the incorporation of residual heat, we used the simulation displayed in Figure 20 to test the estimation framework. Figure 22 provides the resulting estimations of A and σ . To measure the accuracy of the estimations, we computed the absolute errors of between the true model parameters and the estimated model parameters over time, as shown in Figure 23. Referring to the resulting absolute errors, the estimations are less accurate (with a higher absolute error) in the first second of each pulse. The estimations become more accurate (as the absolute errors are low and close to zero) over the duration of each pulse, which verifies the estimation framework.

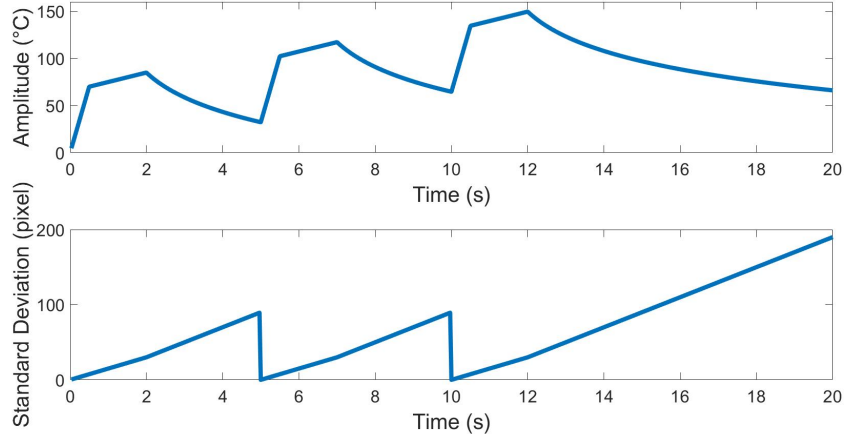


Figure 20: Multiple pulse simulation with residual heat. This plot shows a three-pulse simulation over 20 seconds, such that the duration of every pulse is 5 seconds. Notice that at the end of the third pulse, without the addition of a fourth pulse, at $t = 15$ s, the amplitude continues decreasing and the standard deviation continues increasing.

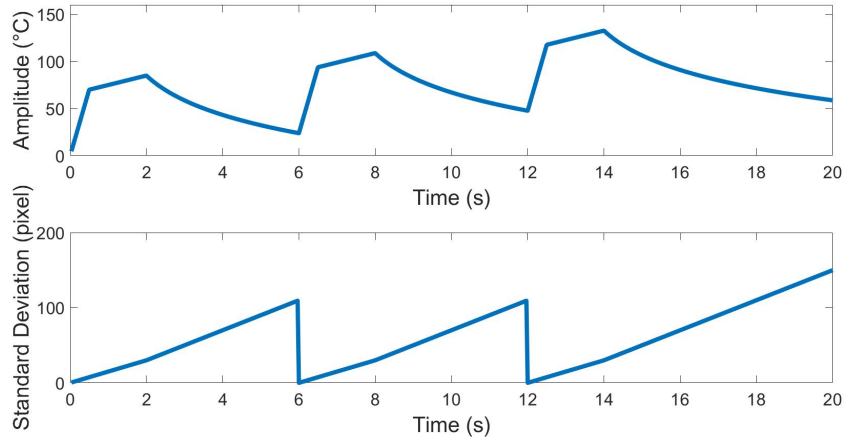


Figure 21: Multiple pulse simulation with residual heat with a different duration of each pulse. The new parameter functions, $A(t, l, p)$ and $\sigma(t, l, p)$, allows more flexibility in simulating temperature data sets for multiple pulses with residual heat. Similar to Figure 20, where $l = 5$, this a three-pulse simulation over 20 seconds, with the duration, $l = 6$.

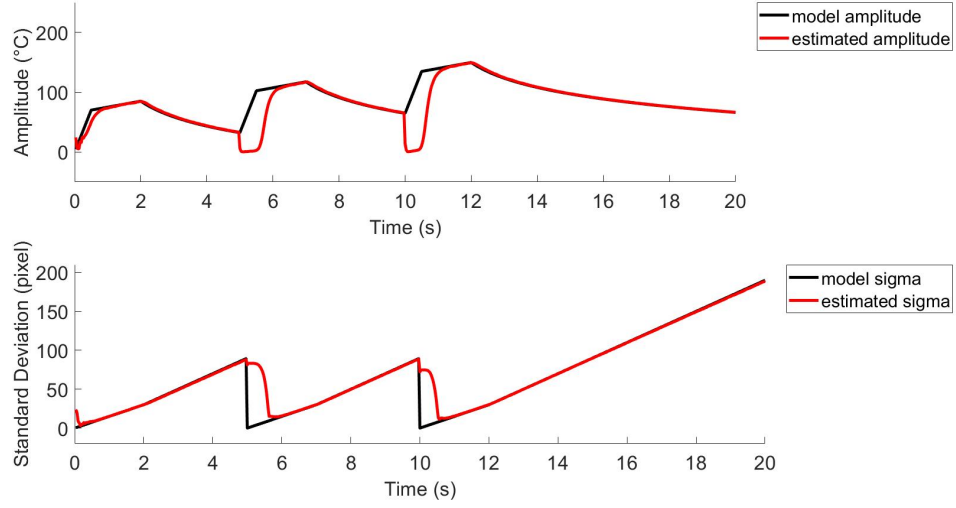


Figure 22: Estimation results from the EnKF framework on a three-pulse simulation with residual heat, as shown in Figure 20. The true model parameters are shown in black and the EnKF mean estimates are shown in red.

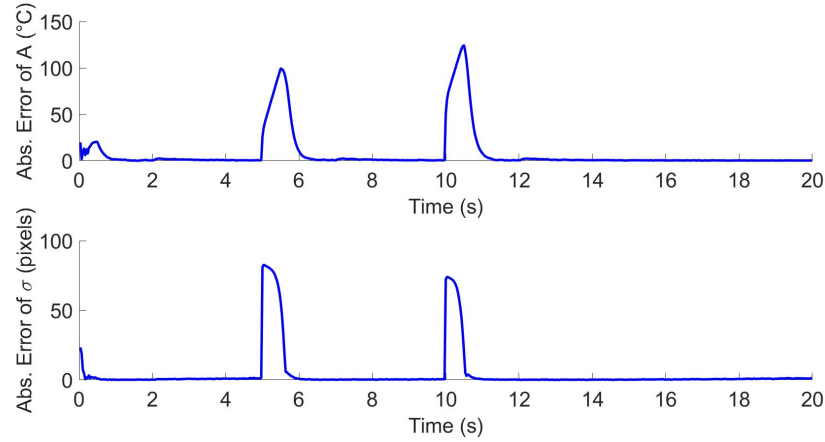


Figure 23: Absolute errors of the A and σ estimations, shown in Figure 22, over time. The absolute error is calculated by computing the absolute difference of the the true model parameters and estimated model parameters over time.

4.2.3 Change in the Center Location

In a real laser procedure, when multiple laser pulses are delivered to the targeted tissue, the surgeon can make small movements and apply the next pulse to a slightly different location. Therefore, the center location of the temperature peak (which is where the laser is delivered) is unlikely to remain the same from pulse to pulse. In the previous data simulations, we assumed the center location, μ_x and μ_y , to be constant during multiple pulses procedure, where $\mu_x = \mu_y = 0$. This assumption works well with the simulation for a single pulse laser delivery. However, this assumption is less realistic during the delivery of multiple laser pulses. Therefore, we decided to incorporate the change of μ_x and μ_y between pulses to generate a more convincing simulation.

Firstly, we assume that the center location changes from pulse to pulse, while it remains constant within each pulse. To incorporate the feature of having μ_x and μ_y vary between pulses, we are adding a random noise term to the base center location at the beginning of each pulse. This random noise terms represent the surgeon's slight movements, and are drawn from a uniform distribution. If the range of the uniform distribution is larger, then it represents larger movements of the center locations. We assume that the movements are small, so we draw the random noise term from a uniform distribution from 0 to 5, and add it to the initial center, $\mu_0 = \mu_0 = 0$. Figure 24 displays the plot of the center location, μ_x and μ_y , in a simulated temperature data set.

Another aspect of the change in the center location is to add a mathematical pattern, like a linear pattern, to μ_x and μ_y . The purpose of this incorporation is to model the case when the laser beam is being applied across the tissue surface, instead of to a certain location. It accounts for the possibility that the center location is moving in both the x and y direction during a single pulse, rather than remaining constant. For example, the thermal camera can be moving linearly in both directions, resulting in the μ_x and μ_y to be linear in the duration of every pulse. Figure 24 displays the plot of the center location of a simulated multiple pulse data set, where μ_x and μ_y shows a linear pattern for every single pulse. The linear patterns for μ_x and μ_y are generated over time t . The slopes of the linear functions are defined as a given value multiplied by the μ_x and μ_y values (which are both equal to $\mu_0 = 0$, plus a random value).

With simulated data on the change of the center location, our next step to test the estimation framework, like what we did in all the previous data simulations. Figure 25 displays the estimations of the μ_x and μ_y parameters. Referring to this figure, although the estimated μ_x and μ_y are oscillating rapidly at the first second, the estimated values eventually become close to the model values. To quantitatively measure the accuracy of the parameter estimations, we computed the absolute error between the true and estimated model parameters over time, displayed in Figure 26. The absolute errors indicates that the estimated μ_x and μ_y are less accurate with higher errors at the start of every pulse and becomes more accurate over the time of every pulse. Therefore, we can conclude that the EnKF framework is verified with the temperature data sets we simulated under the different scenarios we are considering.

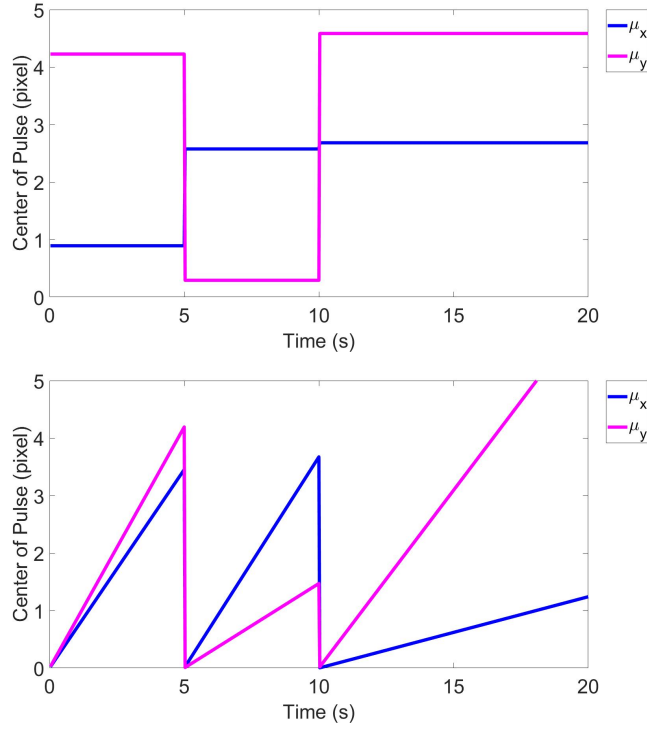


Figure 24: Changing center of pulse: constant within pulse (top) and linear within pulse (bottom). In each plot, μ_x is shown in blue and μ_y is shown in magenta. These plots display the different center locations (μ_x and μ_y) of each additional laser pulse for a three-pulse laser-tissue interaction. In the case at the top, μ_x and μ_y remains constant for the duration of each pulse, but changes its location from pulse to pulse. In the case at the bottom, μ_x and μ_y are linearly changing for the duration of each pulse, and from pulse to pulse.

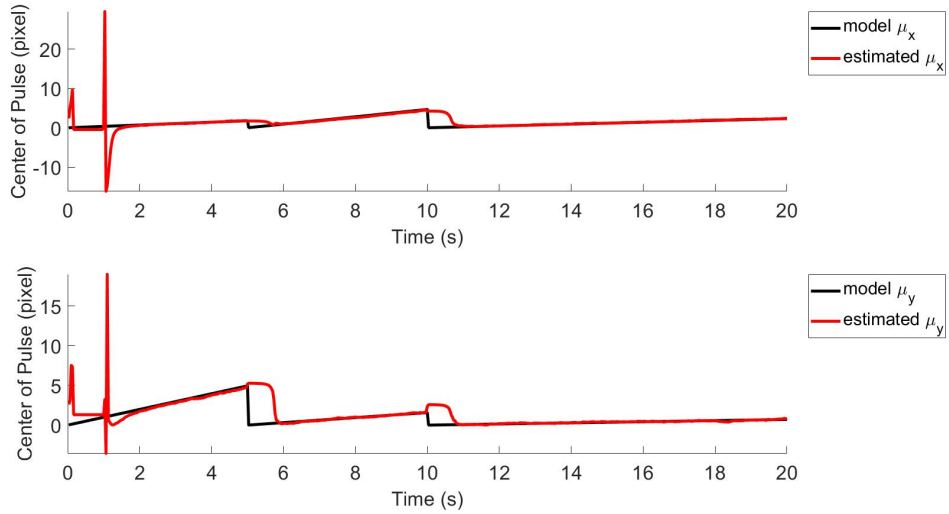


Figure 25: Parameter estimation results for a multiple pulses simulation with changing center location, displayed in Figure 24 (the plot on the bottom). The true model parameters are shown in black and the EnKF mean estimates are shown in red.

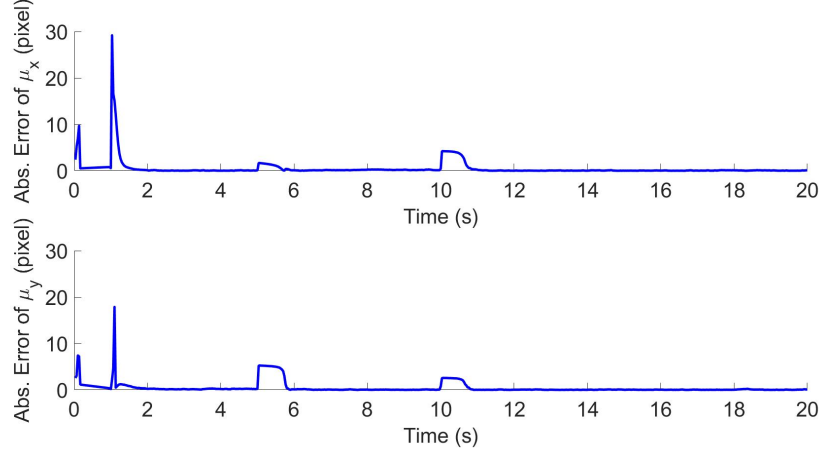


Figure 26: Absolute errors of the A and σ estimations, shown in Figure 25, over time. The absolute error is calculated by computing the absolute difference of the the true model parameters and estimated model parameters over time.

4.3 Estimation Framework Settings with Simulated Data

To develop a parameter estimation framework that produces accurate estimations, it is essential to have an understanding of the effects of different initial inputs of the framework. The initial inputs we consider are as follows:

- the type of the initial prior distribution (for A and σ)
- the observation error (D)
- the model prediction error (C)
- the standard deviation for the random walk (E_A and E_σ)
- the ensemble size (N_{ens})
- the grid size we choose when simplifying observed temperature data in grid pattern

The observations that we will discuss below are based on simulated data, as we run sets of simulated data through the estimation framework to examine the effect of the inputs on the estimation results.

4.3.1 Different Types of Initial Prior Distributions

We are interested in understanding the effects of the type of prior distribution on the estimations of the parameters. We draw a random sample of the amplitude (A) and standard deviation (σ) at $t = 0$ s from the prescribed prior distribution. The types of distributions we tested include the Normal, Exponential, Weibull, Poisson, and Uniform distributions.

We examined the effect of various distributions on the parameter estimations. However, the input parameters differ between distributions. For instance, the input parameters of a Normal distribution are the mean and standard deviation, while the input parameters of a Uniform distribution are the upper and lower endpoints. Therefore, we used different input values to test the various distributions.

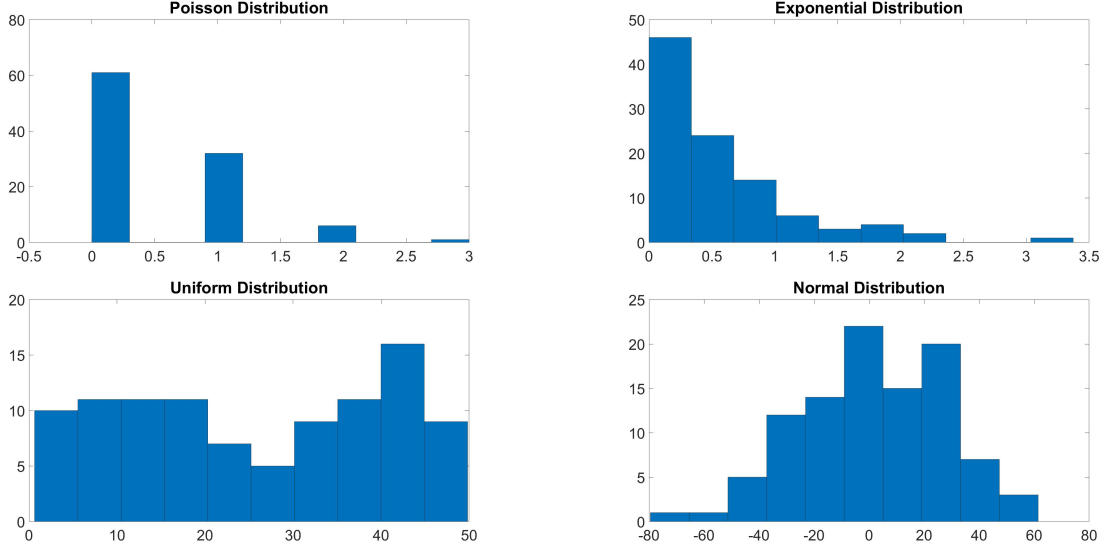


Figure 27: Histograms of the different types of prior distribution of parameters A and σ for estimation. Generated random values for the initial parameters from skewed distributions, Poisson (top left) and Exponential (top right), and symmetric distributions, Uniform (bottom left) and Normal (bottom right), are displayed. For each histogram, 100 random draws are generated.

The type of distribution affects how the values for the initial prior distribution are randomly selected. Although the Poisson and Exponential distributions are both skewed, Figure 27 displays how these distributions differ in selecting random values, such that random values of a Poisson distribution are less scattered. On the other hand, the Normal and Uniform distributions are symmetric. As shown in Figure 27, a Normal distribution is in the shape of a bell curve centered at its mean. Values around or closer to the mean are more likely to be randomly drawn for a Normal distribution. Meanwhile, Uniform distribution results in a rectangular shape that data points are uniformly spread within the lower and upper endpoints, such that random values within the range are equally likely to be drawn.

After running multiple tests on the various types of distributions, we compared the skewed and symmetric distributions. A common result we observed was that the estimated parameters from the skewed distributions tend to oscillate more rapidly during the first second of the simulation before the estimated parameters become close to the model parameters. Greater oscillations indicate a higher possibility of producing inaccurate estimations. Additionally, during a few of the tests, it took the framework a longer duration before it makes accurate parameter estimations. Therefore, we decided to eliminate skewed distributions.

According to the observational results, we concluded that both Normal and Uniform distributions resulted in relatively accurate estimations. The estimated parameters are oscillating less rapidly, and it takes the framework relatively less time to generate accurate estimations. More detailed observational results, for both skewed and symmetric distributions, can be found in Appendix B.1. However, since the initial A and σ are randomly generated values from the distribution, we believe that the Uniform distribution would be a better option because uniformly distributed values are equally likely to be selected.

4.3.2 Observation Error

The observation error (D) reflects how much the estimation framework trusts the observation points. A higher observation error means that the observed data is less trusted. A smaller variance in the data indicates that the observed data is more accurate. Let the processed observed temperature data that we want to estimate be square and the grid size be m , that is $T_{obs} \in \mathbb{R}_{m \times m}$. The observation error is in the

following form:

$$D = \delta_D^2 * \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} \in \mathbb{R}_{m^2 \times m^2} \quad (29)$$

To draw an observational conclusion on the effect of different observation errors on the resulting parameter estimations from the framework, we tested various values for the δ_D^2 . These various values includes 0.005, 0.01, 0.1, 0.5, and 1. Notice that for these tests, the noise level of the simulated temperature data set is 1. A greater value of D results in the less accurate estimated parameters, as the estimations are further away from the model parameters. This is particularly shown in the parameter estimations when $\delta_D^2 = 0.5$, and 1, such that the estimated σ starts to diverge away from the model σ , during the cooling phase of the tissue (after 2 seconds). Furthermore, during both the heating and cooling phases, both tests failed to capture the peak amplitude of the pulse. Meanwhile, when $\delta_D^2 = 0.005$, and 0.01, both the estimated A and σ values are close to those of the model. In fact, the resulting estimations are similar in accuracy and behavior. Therefore, we observed that δ_D^2 values below 0.01 results in better performance. More detailed observational results can be found in Appendix B.2.

4.3.3 Model Prediction Error

The model prediction error (C) shows how much the estimation framework trusts the model states and parameters. A greater model prediction error indicates that the model is less trusted. Let the processed observed temperature data that we want to estimate be square and the grid size be m , that is $T_{obs} \in \mathbb{R}_{m \times m}$. The prediction error is in the following form:

$$C = \delta_C^2 * \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix} \in \mathbb{R}_{m \times m} \quad (30)$$

We tested different values for δ_C^2 , including 0.0001, 0.001, 0.01, 0.05, 0.1, 0.5, and 1. Each test was run ten times and the common results are recorded in Appendix B.3. From the resulting parameter estimations, $\delta_C^2 = 0.001$, 0.001, 0.01, and 0.05 resulted in estimations with similar accuracy, where the estimated A and σ are close to the model A and σ . In fact, even a value of δ_C^2 that is smaller than 0.0001 produced in relatively accurate estimations. On the other hand, with greater values of δ_C^2 , such as 0.1, 0.5, and 1, some runs produced estimations that oscillation more before becoming close to the model parameters, which means that the estimation framework more time to make accurate predictions. After running tests with various inputs for the prediction error, we observed that a greater prediction error can result in less accuracy in the parameter estimations. Therefore, we concluded on small δ_C^2 values, preferably below 0.05.

4.3.4 Parameter Random Walk Deviation

Since the parameters we are estimating are time-varying, we add a drift term to the parameters during the prediction step of the filter to model the change in the parameter values as random walks. For each prediction step, we added a new random variable, E , to the current parameter estimation. The values of E are random variables of a normal distribution with a mean of 0 and a chosen standard deviation [14]. The standard deviation affects how accurate the parameters are estimated. With the addition of the random walk, it is essential to test the effect of different standard deviations of a normal distribution on the resulting estimations.

To make more accurate observations, we ran each test ten times, recorded the common resulting estimations, and ensured that the other input variables of the framework from constant between the various testings. The different values of standard deviation we tested include 0.005, 0.01, 0.025, 0.05, and 0.1. We want the estimated mean parameters to be close to the model parameters, such that the estimations converge. From our observations, we concluded that a relatively greater value of the standard deviation produces more oscillations in the estimation. This is shown in the test when the standard deviations of 0.05 and 0.1, where the amplitude estimation oscillates rapidly during the cooling phase. Also, with a greater standard deviation, the parameter estimations are less likely to converge to the model parameters. This observation is supported by the σ estimations with the standard deviations of 0.05 and 0.1, where the estimated σ diverges after 5 seconds of the single pulse. Meanwhile, lower standard deviations, like 0.005, 0.01, and 0.025, resulted in less oscillation and more accuracy in the estimations, especially in with the value of 0.01. Appendix B.4.1 shows the results of our testings of different standard deviation values. Therefore, we concluded with a standard deviation of around 0.01 would result in better estimations.

When we generated multiple pulse simulations with the change in center, μ_x and μ_y , we made further observations on the standard deviation for the addition of a random walk. As concluded earlier from the observation of a single pulse simulation, a standard deviation of 0.01 results in accurate A and σ estimations. Applying this conclusion, we conducted further analysis on the random walk standard deviation, focusing on the new time-varying parameters, μ_x and μ_y . We started with setting the standard deviation of the random walks to be 0.01. As a result, we observed that the estimated μ_x and μ_y are oscillating around the model μ_x and μ_y during the entire simulated laser-tissue interaction. The A and σ estimations are relatively accurate, as concluded before. Therefore, we decided to test different standard deviations for the random walk of μ_x and μ_y , including 0.001 and 0.02. According to our observation, a lower standard deviation of 0.001 resulted in a more accurate estimation with less oscillation. A greater standard deviation of 0.02 resulted in estimations of low accuracy, which that more oscillation resulted, as compared to the value of 0.01. Detailed testing results can be found in Appendix B.4.2. Therefore, we observed that different standard deviations can be used for the random walk of the time-varying parameters (A , σ , μ_x , and μ_y). We concluded with a standard deviation of 0.01 for A and σ , and a standard deviation of 0.001 for μ_x and μ_y is a possible combination to produce more accurate estimations.

In general, a greater standard deviation of a normal distribution for the random walk is more likely to be able to capture larger changes over time. For example, for the estimation of the amplitude, a greater random walk will better estimate its peak. A lower standard deviation of the random walk will better capture smaller changes in the parameters. For example, for the estimations of the center location, less oscillation was observed.

4.3.5 Input Variables of Uniform Distribution and the Ensemble Size

From our previous observation, we concluded that uniformly distributed random values for the initial parameters (A and σ) will result in more accurate estimations from the framework. As mentioned earlier, various types of distributions require different input parameters. We made observations to conclude on the input values that will result in better estimates. Additionally, we are considering the effect of the ensemble size on the resulting estimations. As described in 3.3, the ensemble size affects the number of ensemble inputs that include the initial guesses of the parameters. Theoretically, a greater ensemble size would result in more accurate estimations. However, a greater ensemble size also requires a longer execution time, which may not be desirable if the accuracy of the estimation does not correspond with the additional run time needed. Therefore, observational tests were run with different input parameters and ensemble sizes. Table 6 provides the descriptions of the tests we conducted (the lower and upper endpoints are the input parameters for the bounds for the uniform distribution). Each test was run five times and the execution time for each run is recorded in Table 7.

Detailed common results of these tests can be found in Appendix B.5. The results of our tests support the fact that a larger ensemble size results in a more accurate estimation. At least one of the five runs for

	Test 1	Test 2	Test 3	Test 4	Test 5
Lower endpoint (A and σ)	0	0	0	0	0
Upper endpoint (A and σ)	50	50	100	100	100
Lower endpoint (μ)	0	0	0	0	0
Upper endpoint (μ)	5	5	5	5	5
Ensemble size (N_{ens})	100	150	250	500	1000

Table 6: Table of the inputs of the estimation framework for the observation tests on various input values of uniform distribution (lower and upper endpoints) and ensemble sizes.

Test No.	Run 1 (s)	Run 2 (s)	Run 3 (s)	Run 4 (s)	Run 5 (s)	Average (s)
Test 1	11.36	10.82	10.88	11.02	11.86	11.19
Test 2	13.53	14.57	16.78	13.92	14.25	14.61
Test 3	20.30	17.68	18.81	18.11	18.51	18.68
Test 4	34.98	36.88	31.49	35.53	34.14	34.60
Test 5	76.35	73.34	78.55	74.41	65.94	73.72

Table 7: Table of execution times for the EnKF framework, with various input values of uniform distribution (lower and upper endpoints) and ensemble sizes. CPU time is measured in seconds. These times are measured using the stopwatch timer (tic, toc function) in MATLAB.

Test 1, Test 2, and Test 3 resulted in more oscillation before the generation of accurate estimations. The delay affects the estimations of the first pulse; the framework fails to capture the first peak amplitude and oscillating during most of the heating phase. Meanwhile, we observed that all five runs for Test 4 resulted in relatively accurate estimations, the framework settled to trust-able estimations in a reasonable time (within the first 2 seconds). From our observations for Test 5, which has a larger ensemble size, we concluded that the resulting estimations were similar to that of Test 4. We did not perceive a great improvement in accuracy to account for the extended execution time. As a result, we believe the input variables and ensemble size (of 500) for Test 4 will provide us with relatively accurate estimations at a reasonable time. Appendix B.5 provides details on our observations.

4.3.6 Choice of Grid Size in Data Pre-Processing

As described in Chapter 3, the set of observed temperature data is large, which results in a long execution time to estimate the time-varying parameters. To address this problem, we need to simplify the observed temperature data set by selecting certain data points from the original data set in a grid pattern. Although the approach to generate a simplified data set improves the duration of the estimation framework, it will lead to a loss of data. In this section, we test different grid sizes when pre-processing the temperature data. We aim to find a desirable balance between the execution time of the framework and the level of accuracy of the resulting estimations. More accurate estimations can provide more correct feedback of the tissue type. Meanwhile, a shorter execution time of the estimation framework will provide more real-time feedback.

We perform observational tests with a set of observed data simulated from MATLAB, as described in Chapter 3, and each instantaneous temperature surface $T_{obs} \in \mathbb{R}_{100 \times 100}$. First, we simplify the observed data set in a grid pattern, with grid sizes of 10, 20, and 50, which are factors of 100. Then, we estimate the simplified data with the EnKF framework. The comparison on the efficiency of different grid sizes involves the evaluation of two aspects: the accuracy of the estimations and the length of the execution time.

In terms of the execution time, we performed five tests for each of the grid sizes (10, 20, and 50) with the same set of simulated sequential temperature data, and measured the execution time of each test. We observed that the grid sizes of 10 and 20 resulted in shorter execution time. However, the grid size of 50 required a longer duration, longer than 120 s. For a grid size of 10, the execution time of the estimation

Test No.	Grid Size 10 (s)	Grid Size 20 (s)	Grid Size 50 (s)
Test 1	0.29	1.69	130.92
Test 2	0.32	1.87	122.49
Test 3	0.34	1.73	141.03
Test 4	0.28	1.70	125.22
Test 5	0.33	1.81	125.10
Average	0.31	1.76	128.95

Table 8: Execution time table of estimating the pre-processed temperature data with different grid size.

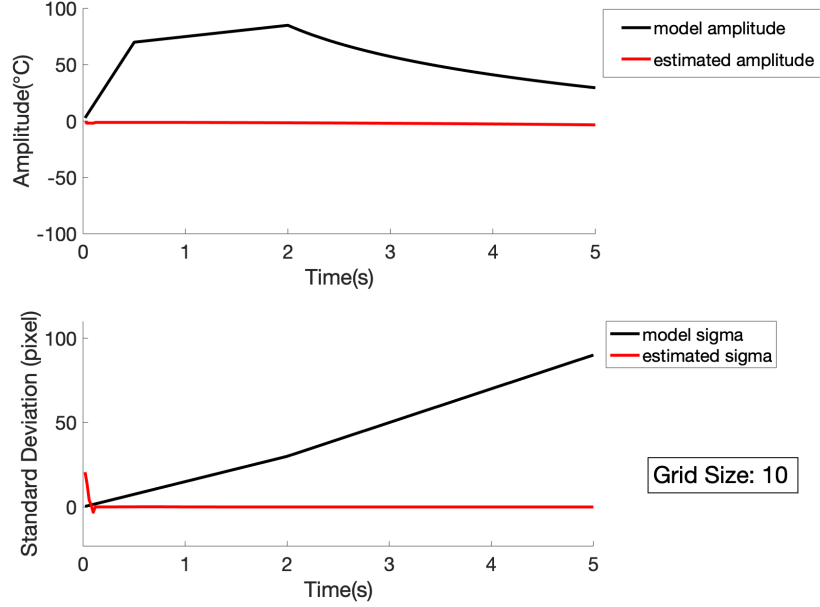


Figure 28: This plot shows the model parameters (in black) and the estimated parameters (in red), with the grid size of 10. The estimations are the mean values of the posterior density with respect to the temperature data, simplified to $\mathbb{R}_{10 \times 10}$ in grid pattern. The estimated parameters diverge from the true model parameters, indicating poor performance.

framework was 0.31 s, on average. For a grid size of 20, the estimation framework took an average of 1.76 s to complete. For a grid size of 50, the duration of the estimation framework was 128.95 s, on average, which is undesirable. If we want to provide online feedback to the surgeon in the future, this execution time would be too long. As a result, the grid sizes of 10 or 20 generated parameter estimations in a reasonable duration. Table 8 displays the details of each test.

In terms of accuracy, we observed that choosing grid sizes of 20 and 50 resulted in better performances. However, the simplified data set with grid size of 10 resulted in less accurate estimations, such that the estimated parameters diverge from the true model parameter values, as shown in Figure 28. This indicates that information is lost during the data simplification process. Based on the accuracy of the estimations, we observed that grid sizes of 20 or 50 is more desired. Figures 28, 29, and 30 displays the estimation results with grid sizes of 10, 20, and 50, respectively.

As a result of our observational tests on the effect of various grid sizes, we concluded that a grid size of 20 will result in more accurate estimations in a reasonable time. A simplified 20×20 grid provides a smaller data set that is representative of the temperature evolution, and estimations can be made using the EnKF framework within a 2 s execution time.

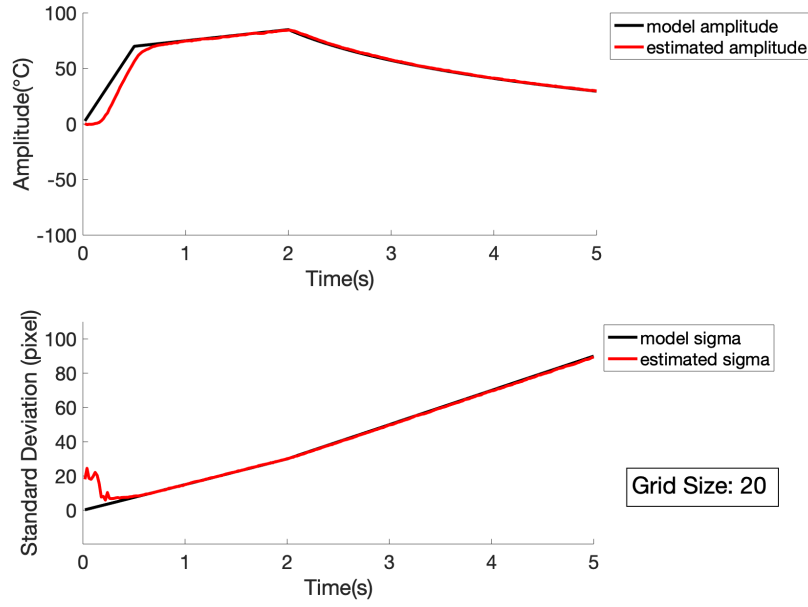


Figure 29: This plot shows the model parameters (in black) and the estimated parameters (in red), with the grid size of 20. The estimations are the mean values of the posterior density with respect to the temperature data, simplified to $\mathbb{R}_{20 \times 20}$ in grid pattern. The estimated parameters are close to the model parameters, indicating a good performance.

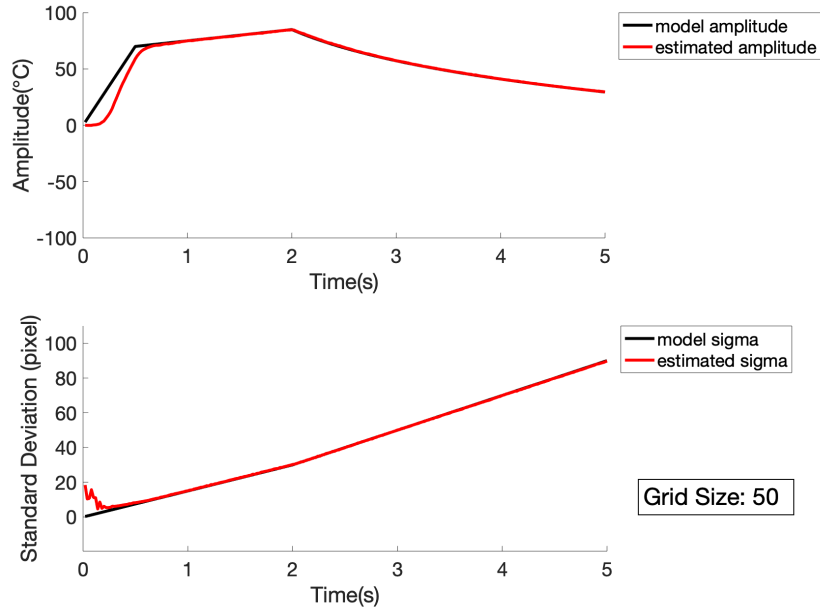


Figure 30: This plot shows the model parameters (in black) and the estimated parameters (in red), with the grid size of 50. The estimations are the mean values of the posterior density with respect to the temperature data, simplified to $\mathbb{R}_{50 \times 50}$ in grid pattern. The estimated parameters are close to the model parameters, indicating a good performance.

4.4 Classification Results

The following section details the results of the classification models in determining the type of simulated data. We chose to test two classification models, the Linear Discriminant Analysis (LDA) and Support Vector Machine (SVM), because of their easy training cost and interpretation of results. The classification experiments can be summarized in the list below. Each experiment is meant to test the robustness of the model with changing simulation parameters with the eventual goal of creating a reliable and robust classification model that allows the distinguishing of real experimental data. Listed below, are three different experiments that were executed to test the classification model.

- Experiment #1: Modifying the maximum amplitude difference between types of simulated data. These experiments tested the classification model's ability to distinguish between types of simulated data with differences in maximum amplitude (A_{max}).
- Experiment #2: Modifying the test data to have maximum amplitudes different from the training data set. Using data with different A_{max} to test a model trained with a separate A_{max} .
- Experiment #3: Modifying the noise on the maximum amplitude A_{max} during the generation of simulated data.

4.4.1 Amplitude Difference in Classification

Through these experiments, we would like to determine what the minimum difference between thermal responses the model is able to correctly classify. For these experiments, Type A data, was kept at maximum amplitude of 170 °C, while Type B data was varied between 175, 180, 200, and 220 °C. The result of these variations in maximum amplitude can be seen in the derived features of the classification models in Figure 31. The smaller the difference in maximum amplitude between the two data types, the more the two clusters of data approach each other. Two clusters that are located within each other have a high covariance and are difficult to distinguish. This fact is represented in Figure 32 which shows a reduction in accuracy of classification with a smaller difference in amplitude between Type A and B. The confusion matrices in Figure 32 represent the performance of an LDA classification model trained with $N = 1000$ and tested with $N = 100$ newly generated data points. Further parameters for simulated data can be found in Section 3.4. The classification is accurate, above 90%, for differences in A_{max} of 30 °C and above. Classification accuracy degrades rapidly below this threshold, with 10 °C difference at a 77% accuracy, and a 5 °C difference with a 52% accuracy.

4.4.2 Robustness of the Classification Model

The second set of experiments focused on adding noise to the maximum amplitude of the training and testing data. Resulting in an overall increase in spread of the two clusters within the feature space increases as seen in Figure 35. Adding noise to the maximum amplitude has a similar effect to decreasing the maximum amplitude difference between two types of data. As seen in Figure 36, the accuracy of a model trained with increased noise, and tested on data without noise on the amplitude, is comparable to a model trained without noise. Adding noise in this manner, decreases the accuracy of classification, comparable to decreasing the difference in maximum amplitudes between tissue types.

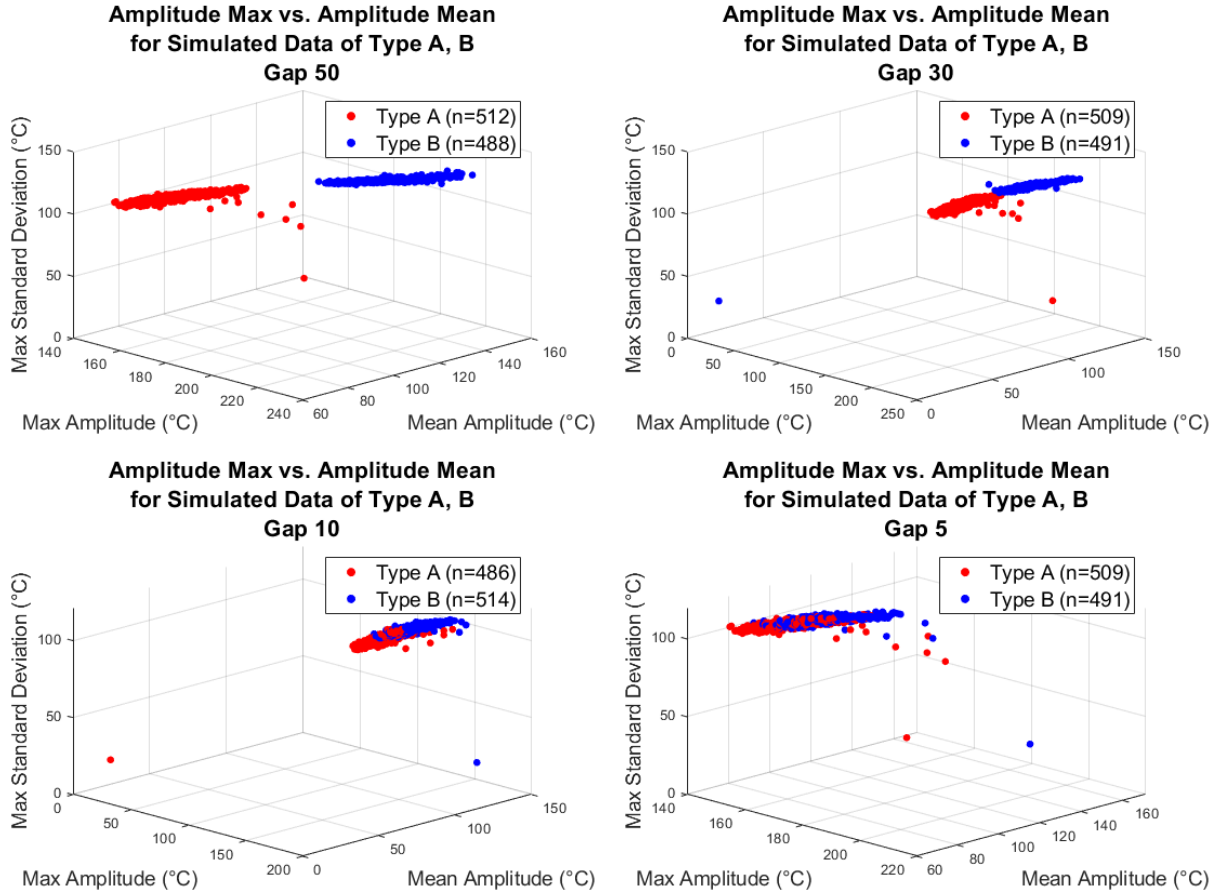


Figure 31: The effect of a difference in maximum amplitude on the known classification features. 1000 simulated data points were generated and input into the Estimated Kalman Filter from which classification features are extracted and plotted. The greater the space between the two clusters, the more accurate classification is. Therefore, a higher difference in maximum amplitude between simulated types of tissue will result in more effective classification.

**LDA model trained with 50°C amplitude difference,
tested with 50 °C difference (N = 100)**

		Generated Type (True Label)		
		Type A	Type B	Total Predicted
Predicted Type (Predicted Label)	Type A	42%	0%	42
	Type B	0%	58%	58
	Total Generated	42	58	100%

**LDA model trained with 30°C amplitude difference,
tested with 30 °C difference (N = 100)**

		Generated Type (True Label)		
		Type A	Type B	Total Predicted
Predicted Type (Predicted Label)	Type A	47%	5%	52
	Type B	1%	47%	48
	Total Generated	48	52	94%

**LDA model trained with 10°C amplitude difference,
tested with 10 °C difference (N = 100)**

		Generated Type (True Label)		
		Type A	Type B	Total Predicted
Predicted Type (Predicted Label)	Type A	43%	17%	60
	Type B	6%	34%	40
	Total Generated	49	51	77%

**LDA model trained with 5 °C amplitude difference,
tested with 5 °C difference**

		Generated Type (True Label)		
		Type A	Type B	Total Predicted
Predicted Type (Predicted Label)	Type A	28%	19%	47
	Type B	27%	26%	53
	Total Generated	55	45	54%

Figure 32: Accuracy of the Linear Discriminant Analysis (LDA) classification model with differences in maximum amplitude. Each matrix represents 100 data points of simulated data generated and identified by the machine learning model. The y-axis represents the labels predicted by the LDA model and the x-axis the known correct labels. The result is that the top-left and bottom-right elements represent correct predictions, shown in green. Due to there being two elements, a classification accuracy of 50% is the lowest possible.

SVM model trained with 50 °C amplitude difference, tested with 50 °C difference (N=100)

		Generated Type (True Label)		
		Type A	Type B	Total Predicted
Predicted Type (Predicted Label)	Type A	46%	0%	46
	Type B	0%	54%	54
	Total Generated	46	54	100%

SVM model trained with 30 °C amplitude difference, tested with 30 °C difference (N=100)

		Generated Type (True Label)		
		Type A	Type B	Total Predicted
Predicted Type (Predicted Label)	Type A	56%	0%	56
	Type B	0%	44%	44
	Total Generated	56	44	100%

SVM model trained with 10 °C amplitude difference, tested with 10 °C difference (N=100)

		Generated Type (True Label)		
		Type A	Type B	Total Predicted
Predicted Type (Predicted Label)	Type A	42%	13%	55
	Type B	7%	38%	45
	Total Generated	49	51	80%

SVM model trained with 5 °C amplitude difference, tested with 5 °C difference (N=100)

		Generated Type (True Label)		
		Type A	Type B	Total Predicted
Predicted Type (Predicted Label)	Type A	28%	19%	47
	Type B	27%	26%	53
	Total Generated	55	45	54%

Figure 33: Accuracy of the Support Vector Machine (SVM) classification model with differences in maximum amplitude. Each matrix represents 100 data points of simulated data generated and identified by the machine learning model. The y-axis represents the labels predicted by the SVM model and the x-axis the known correct labels. The result is that the top-left and bottom-right elements represent correct predictions, shown in green. Due to there being two elements, a classification accuracy of 50% is the lowest possible.

**LDA model trained with 30 °C amplitude difference,
tested with 50 °C difference (N=100)**

		Generated Type (True Label)		
		Type A	Type B	Total Predicted
Predicted Type (Predicted Label)	Type A	48%	1%	49
	Type B	0%	51%	51
	Total Generated	48	52	99%

**LDA model trained with 10 °C amplitude difference,
tested with 50 °C difference (N=100)**

		Generated Type (True Label)		
		Type A	Type B	Total Predicted
Predicted Type (Predicted Label)	Type A	54%	9%	63
	Type B	0%	37%	37
	Total Generated	54	46	91%

**LDA model trained with 50 °C amplitude difference,
tested with 30 °C difference (N=100)**

		Generated Type (True Label)		
		Type A	Type B	Total Predicted
Predicted Type (Predicted Label)	Type A	24%	0%	24
	Type B	25%	51%	76
	Total Generated	49	51	75%

**SVM model trained with 50 °C amplitude difference,
tested with 30 °C difference (N=100)**

		Generated Type (True Label)		
		Type A	Type B	Total Predicted
Predicted Type (Predicted Label)	Type A	41%	0%	41
	Type B	10%	49%	59
	Total Generated	51	49	90%

Figure 34: Accuracy of the Support Vector Machine (SVM) and Linear Discriminant Analysis (LDA) classification model with differences in the maximum amplitude of testing data. The top row shows LDA classification models trained with a lower amplitude gap and tested with a higher amplitude gap. The plot on the bottom displays a model that was trained with a 50 °C gap, but tested with a 30 °C gap, resulting in a decrease in accuracy. Also note the bottom-left figure displays the confusion matrix of the SVM classification model which boasts a higher accuracy than the LDA model.

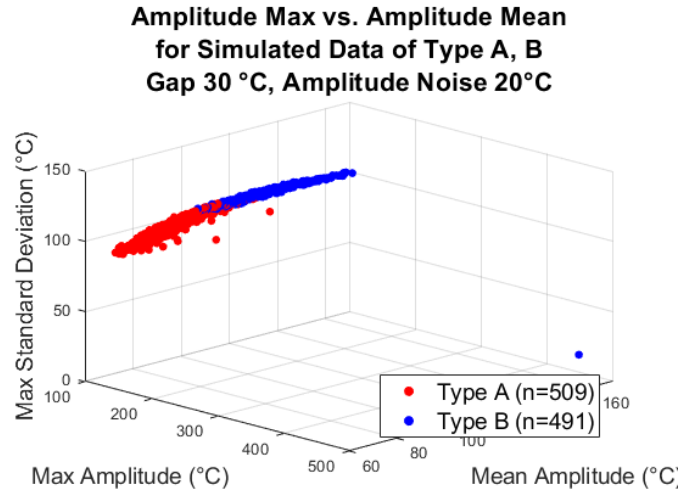


Figure 35: Effect of noise on the classification feature space. Noise on the maximum amplitude A_{max} results in a wider spread for both clusters.

**LDA model trained with 30 °C amplitude
difference and 20 °C noise, tested with 30 °C
difference, and 10 °C noise (N=100)**

		Generated Type (True Label)		
		Type A	Type B	Total Predicted
Predicted Type (Predicted Label)	Type A	49%	0%	49
	Type B	1%	50%	51
	Total Generated	50	50	99%

**LDA model trained with 30°C amplitude
difference and 20°C noise, tested with 30°C
difference, and 20°C noise (N=100)**

		Generated Type (True Label)		
		Type A	Type B	Total Predicted
Predicted Type (Predicted Label)	Type A	45%	7%	52
	Type B	12%	36%	48
	Total Generated	57	43	81%

Figure 36: Accuracy of the Linear Discriminant Analysis (LDA) classification model with trained with noise on the maximum amplitude. The left confusion matrix shows that a classification model trained with an absolute noise of 20 °C, but tested with less noise of 10 °C is still accurate. On the right, a confusion matrix showing a model trained with noise on the amplitude, and tested with the same data.

5 Development of a Bench-Top Experimental Setup

5.1 Data Collection Materials and Methods

Besides using synthetic data to test the hypothesis of the mathematical models and predictions, we also created an experimental setup (see Figure 37) to collect thermal data in a controlled environment. The setup includes a Sharplan 30C CO₂ surgical laser to fire pulses at phantom tissue consisting of Agar [16], a FLIR A655sc infrared thermal camera, (FLIR Systems Inc., Wilsonville, OR) to collect thermal data, and a Franka Emika PANDA robotic arm (Franka Emika GmbH, Germany) to support and maneuver the laser fiber. A mount was 3D printed to hold the laser fiber in place on the robotic arm to ensure the distance between the laser and tissue was consistent between trials. The FLIR camera is mounted on a Noga arm (Noga Engineering & Technology Ltd, Israel), clear of the path of the robotic arm, but with a direct view of the entire tissue sample.

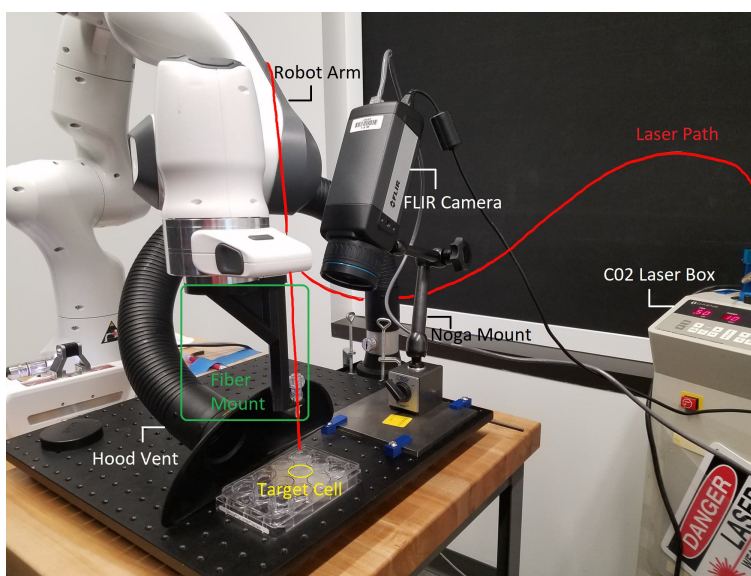


Figure 37: The experimental data collection setup includes: a laser fiber (in red) passes from the CO₂ laser control box into a mount (boxed in green) on the robotic arm which is positioned over Agar in the current target cell (circled in yellow), a FLIR camera is held in place, facing the target cell, by a Noga arm mount which is secured to a steel plate on the workbench, and a hood vent is located near the site of the laser-tissue interaction for fumes produced by the laser firings.

5.1.1 Considerations on the Design of the Experimental Setup

The design of the experimental setup shown in Figure 37 was made based on the following considerations: the experimental data should represent the temperature of the surface of the phantom tissue when fired upon by a laser; the experiment should be run for numerous different trials, and output the same expected results from the mathematical model. To achieve this, the experimental setup needs to be consistent between each firing, such that the experimental data collected should appear the same between repeated trials. The laser fiber, camera and agar should be fixed in a reproducible configuration to ensure the collected data sets have minimal variable changes. For example, the distance between the laser fiber and the surface of the agar should not vary. Other environmental variables such as temperature of the agar and the container should be kept as consistent as possible. The laser settings need to be recorded for every trial and made available to the individuals processing the data using the mathematical model, and subsequently analysing the results;

lastly, the experimental setup needs to have a degree of automation that will allow for collection of all of the data in a short amount of time, allowing for batch data collection.

5.1.2 Equipment

In the following, we describe the pieces of equipment used in the experimental setup (refer to Fig. 37). The Sharplan CO₂ Surgical Laser is able to output an infrared laser beam (wavelength: 10.6 μm) at a power between 1 - 30 Watts (W). The laser beam is delivered to the target area with a flexible fiber. For the purpose of this project, the team set this laser to continuous wave operation mode and single pulse exposure mode, meaning that the laser was fired continuously for a specific duration of time whenever a firing signal was received. In these modes, the laser can produce power between 1 - 4.5 W for a duration of 0.05 - 1.0 seconds, or 5 - 30 W for a duration of 0.01 - 1.0 seconds. A control box was utilized in order to have computer-controlled activation of the the surgical laser.

The FLIR A655sc infrared thermal camera can provide a 640 \times 480 pixel video feed at a maximum frame rate of 100 frames per second (fps). For the purpose of this project, the team used a lens with a 25 degree field of view at a frequency of 50 Hz and a variable recording duration between 6 and 8 seconds. The data from the thermal camera can be accessed using MATLAB with the GigE using the Image Acquisition Tool. This library allows for control of the settings of the thermal camera and capture of thermal data.

The laser fiber is 1.8 mm in diameter, and carries the laser beam to the tissue sample. A 3D printed mount was created to attach onto the end-effector plate of the PANDA robotic arm and securely hold the laser fiber in place. The Franka Emika PANDA Robotic Arm is a 7 Degree of Freedom (DoF) robotic arm that has a pose repeatability of 0.1 mm and a path deviation of less than 1.25 mm. Through existing ROS and libfranka code, the robot position of the end effector can be controlled by an Xbox controller.

Additional materials used in this experiment included a Noga MG61003 Arm with Magnetic Base and Fine Adjust. With the help of a camera adapter, it allowed for the A655sc thermal camera to be securely mounted to the end. Using the adjustments on the arm, the camera can be pointed at a specific target accurately. To mount the magnetic base of the Noga arm, a magnetic fixture had to be designed. A magnetic steel plate was ordered and 3D printed mounts were made to fix it to the table.

5.2 Running Experiments to Collect Real Data

5.2.1 Tissue Targets

In order to conduct a more controlled experiment, we chose to mix Agar powder with water to form a solution for a gel that has properties similar to biological tissue, but results in a more uniform surface to experiment upon. To better mimic the laser absorption properties of real tissue, the following tissue parameters were varied in each experiment: water content (% volume), and the amount of red food dye (mL).

In order to create these Agar samples, the following steps were followed: water (preferably distilled water) is heated until it boils, before mixing in Agar powder as determined by the Agar concentration for the given experiment. If the Agar sample requires red food dye, it is also added in with the Agar powder to be mixed into the Agar solution. This solution is then poured into a 12 well Corning™ Costar™ Flat Bottom Cell Culture Plate and moved into the lab's refrigerator to chill for 30 minutes and solidify into a gel. Once the Agar is finished setting, it is then taken out and left to adjust back to room temperature before experimentation occurs.

While Agar gel was chosen for initial testing, more realistic biological tissue such as chicken breast would be a plausible next step in generating more concrete observations on the reliability of the predictions made

by the mathematical models. However, for the beginning stages of this project, real biological tissue is too variable and complex to conduct our experiments on.

5.2.2 Experimental Protocol

In the following, we describe the experimental protocol for the laser experiments. Before experimental data collection, we had to first prepare the Agar targets that would be used for the experiments, as described in Section 5.2.1. When testing, the initial temperature of the tissue samples before firing was measured to confirm test samples had reached the same temperature as the base. Laser parameters were set on the CO₂ laser control box as outlined for each experiment, and then the laser was set to its ready state. To start collecting experimental data, the process was activated on the computer which sent a signal to the laser to fire, while it simultaneously started data capture with the FLIR camera. After confirming the experimental data capture was complete, we prepared for new experimental data by renaming the file for the new trial and relocating the laser fiber to the next target cell.

5.3 Preliminary Data Collection

Initial testing was conducted in order to get familiar with the timing of preparation, to identify the various control variables, and to become familiar with the flow of data capture through the physical setup and into the data collection software. The reliability and repeatability of these initial tests are uncertain, as the documentation and preparation processes for important control variables such as height of the laser from the Agar gel, distance from the FLIR camera to the target cell, and starting temperature of the Agar were not consistently monitored, noted, nor replicated between experiments.

The preliminary data was captured using the setup depicted in Section 5.1. The first set of data varied laser power and pulse duration on a consistent Agar concentration of 2%. Power settings ranged between 5W - 20W with laser pulse duration varying between 0.1 seconds - 0.3 seconds. A few samples contained 10mL of red food dye (added to a 50mL Agar solution), though most firings were conducted on samples containing no food dye at all. This data set still required manual firing and data collection, as we had not yet incorporated the control system for synchronous firing and data collection. As such, the timing of the multiple pulses for each trial in this experiment was not reliable nor repeatable. Distance measurements between any devices and the Agar samples were absent for this data set. Recorded starting temperature of the Agar before beginning firings was also absent during this data set.

The second set of data implemented the control system, thus the data collection process ran much smoother during this experimental data set. This data set also varied power and pulse duration with a consistent Agar concentration of 2%. Power settings were at 10W and 20W, and for each power setting, pulse duration ranged between 0.1 seconds - 0.7 seconds. No food dye was added to the Agar samples for this experiment. This experiment captured a single pulse per trial instead of multiple pulses. Distance measurements in inches were taken between the FLIR camera and the target cell (7.5 inches), and the FLIR camera and the laser tip (7.25 inches). However, the height of the Agar in each cell varied, and not all heights were recorded. Recorded starting temperature of the Agar before beginning firings was absent during this data set.

The final experiment conducted was meant to vary Agar properties instead of laser settings. Agar concentration was 2% for the *thinner* Agar, and 6% for the *thicker* Agar. Laser power was consistent at 10W and laser duration was maintained at 0.5 seconds. Each trial for this experiment captured a single pulse, fired through the control system. Distance measurements in millimeters were taken between the laser tip and the target cell (13mm - 18mm), and the FLIR camera and the laser tip (19.5mm). However, the height of the Agar in each cell varied, and not all heights were recorded. Before beginning firings, the chilled Agar was left to adjust to room temperature, and the temperature of the Agar was recorded before any firings occurred. However, no further temperatures were recorded between trials for this data set.

5.4 Experimental Data Analysis

With the experimental results we collected, we need to build an analysis application to examine the data. We built the analysis application with MATLAB App Designer, and in this section we will introduce the components and features of this application. This application can be used to set up the EnKF framework and estimate experimental data. Figure 38 and Figure 39 display the user interface of the analysis application, which consists of three panels: left panel, middle panel, and right panel.

With the analysis application, we can use the left panel to parse the original experimental data and save the parsed data to the disk. First, we need to fill in the input fields on the left panel, then click ‘Visualize’, and adjust the input values until we get an appropriate set of input values, then click ‘Save’ to save the processed data. Further details on the input fields shown in the left panel are as follows:

- **Raw Temperature Frames:** This is where we input the absolute filename (in text) of the raw data that we want to parse. There are two ways that we can input this information, we can either type it in or click ‘Browse’, then locate and select the files in the pop-up window.
- **Center (X) and Center (Y) (pixel):** These two fields consumes an integer value that represents the center position of the targeted parsed data. We can adjust the center position by referencing the ‘Visualization of Processed Data’, located in the same panel.
- **Width/Length (pixel):** This field consumes an integer value and determines the range of the processed data. To ensure that there is no information missing, we should choose a range such that the entire temperature peak can be included in the parsed data. The default value here is 100.
- **Grid Size:** This field consumes an integer value and determines the number of data points we draw from the chosen range in a grid pattern. We need to choose a number such that the width and length can be divided by the grid size. The grid size is recommended to be below 20 for faster results in the estimation.
- **New Frequency (frames per second):** This field consumes an integer value and determines the new frequency of the sequential temperature data. The new frequency must be positive, while the original frequency needs to be divisible by the new frequency.
- **Beginning Frame and Ending Frame:** Since the raw data sequence may include multiple pulses and spare frames, the user needs to specify the starting and ending frames to ensure that the only frames kept are those that they needed.
- **Baseline (°C):** We need the baseline of the Gaussian model to be 0 to apply the model. However, the base temperature of the temperature profile is usually not equal to 0. Therefore, we need to specify a baseline value (as an integer value) to be subtracted from the temperature sequence. Then, we need to manually decide on the baseline of the experimental data.
- **Processed Data Folder and Processed Filename:** These fields specify where the processed temperature profile will be saved (in text). The default target directory is where the raw data file locates and the default filename is the name of the raw data file with ‘processed_’ prep-ended to it.

Once the thermal data has been pre-processed, the next step is to set up the augmented Ensemble Kalman Filter, which can be done using the middle panel of the interface. Click the ‘Start’ button to start the estimation of the data. The sequential estimated temperature states and parameters are demonstrated on the right panel. With this analysis application, we can treat the amplitude, the standard deviations (in both the x and y directions), and the center position (in both the x and y directions) as time-varying parameters. The following gives an explanation of each field in the middle panel.

- **A Mean (°C):** We draw random samples of the initial amplitude (A) from a Normal prior distribution. This field consumes a number and defines the mean of the Normal prior distribution of A .

Raw Data Processing

Raw Temperature Frames

Centre(X) (pixel)

Centre(Y) (pixel)

Width/Length(pixel)

Grid Size

New Frequency (fps)

Beginning Frame

Ending Frame

Baseline (°C)

Processed Data Folder

Processed Filename

Visualization of Processed Data

Parsed Data

A Mean (°C)

A Std Deviation (°C)

Sigma Mean (pixel)

Sigma Std Deviation (pixel)

Mu Mean(x) (pixel)

Mu Std Deviation(x) (pixel)

Mu Mean(y) (pixel)

Mu Std Deviation(y) (pixel)

Ensemble size

D (°C)

C (°C)

Sigma Random Walk Deviation (pixel)

Amplitude Random Walk Deviation (°C)

Mu Random Walk Deviation (pixel)

☐ Enable Reatime Animation

Export Name

Export Directory

☐ Export Video

Estimation Filename

Figure 38: This figure shows the left panel and the middle panel of the application. The left panel allows the user to select and parse raw data collected from the thermal camera, and the middle panel allows the user to set up the output settings and EnKF settings, including initial prior distributions, ensemble size, observation error covariance, prediction error and random walk deviations.

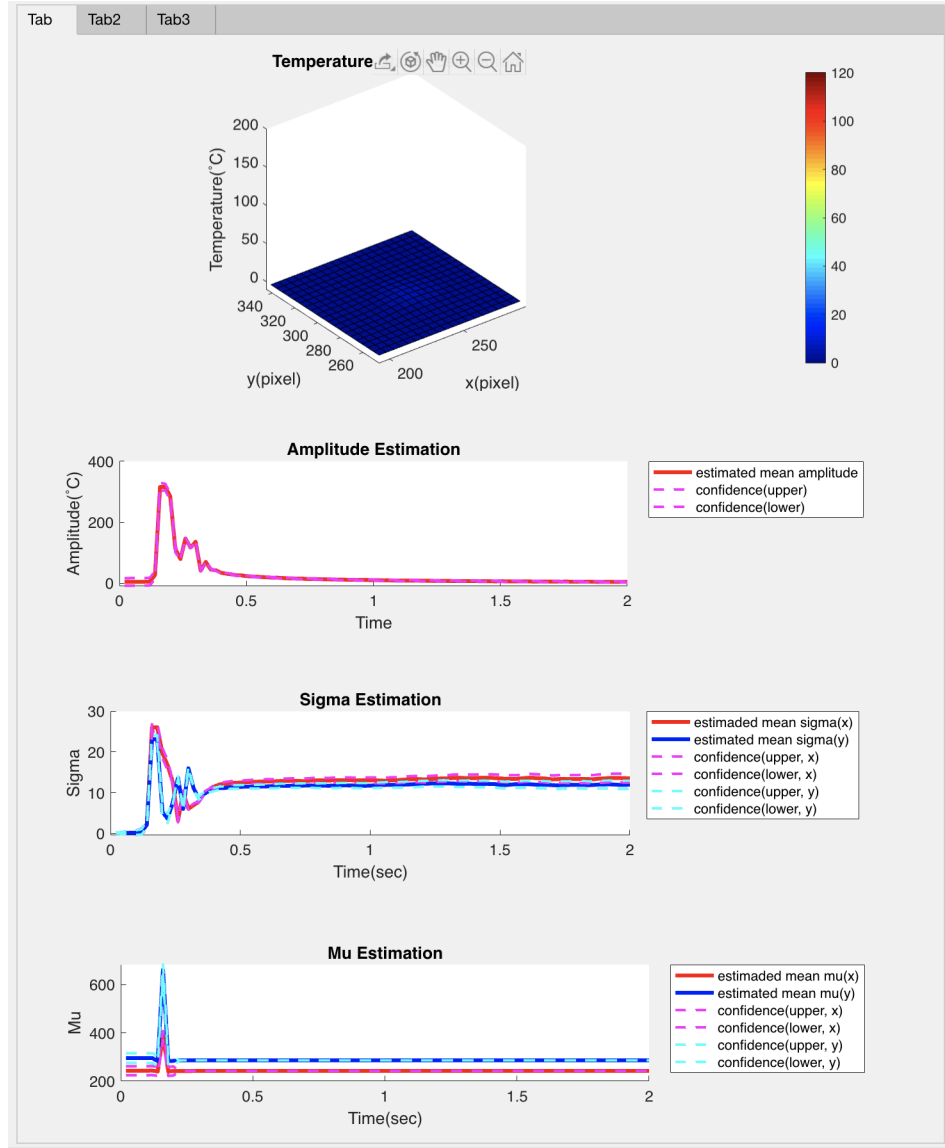


Figure 39: The right panel in the analysis application displays the estimation results, including temperature states, and the parameter estimates, which are given by the posterior ensemble mean.

- A Std Deviation (°C): This field consumes a number and defines the standard deviation of the initial Normal prior distribution of A .
- Sigma Mean (pixel): We draw random samples of the initial standard deviation (σ_x and σ_y) from a Normal prior distributions. Initially, we hypothesize that the spatial spread of the temperature data is symmetric in the x and y directions. This field consumes a number and defines the mean values of the initial Normal prior distributions of σ_x and σ_y .
- Sigma Std Deviation (pixel): This field consumes a number and defines the standard deviation values of the initial Normal prior distributions of σ_x and σ_y .
- Mu Mean (x) (pixel): We draw random samples of the initial center position in the x direction (μ_x) from a Normal prior distribution. This field consumes a number and defines the mean value of the initial Normal prior distributions of μ_x .
- Mu Std Deviation (x) (pixel): This field consumes a number and defines the standard deviation of the initial Normal prior distribution of μ_x .
- Mu Mean (y) (pixel): We draw random samples of the initial center position in the y direction (μ_y) from a Normal prior distribution. This field consumes a number and defines the mean value of the initial Normal prior distributions of μ_y .
- Mu Std Deviation(y) (pixel): This field consumes a number and defines the standard deviation of the initial Normal prior distribution of μ_y .
- Ensemble Size: This field consumes an integer value and defines the ensemble size for the EnKF framework. With a larger ensemble size, the estimations will take longer to generate, but the performance will be better.
- D (°C): This field consumes a number and defines the δ_D^2 of the observation error covariance used in the EnKF framework. For a more accurate set of observed data we should select a smaller δ_D^2 .
- C (°C): This field consumes a number that defines the δ_C^2 of the model prediction error that we use in the EnKF framework.
- Sigma Random Walk Deviation (pixel): This field consumes a number and defines the random walk deviation that we use to predict σ_x and σ_y at each prediction step. If σ_x and σ_y are treated as constant, this field should be set to 0. If σ_x and σ_y are time-varying, the faster these parameters change in terms of time, we choose a larger random walk deviation for them.
- Amplitude Random Walk Deviation (°C): This field consumes a number and defines the random walk deviation that we use to predict the amplitude (A) at each prediction step. The faster the amplitude changes in terms of time, we choose a larger random walk deviation for A .
- Mu Random Walk Deviation (pixel): This field consumes a number and defines the random walk deviation that we use to predict μ_x and μ_y at each prediction step. If we assume that the center of the temperature remains constant, this field should be set to 0. If we assume that the center is time-varying, we should input a positive number. The faster μ_x and μ_y change in terms of time, we should choose a larger random walk deviation for them.
- Enable Real-time Animation: By checking this checkbox, the analysis application will provide an animated visualization of the estimated parameters and the temperature states, located in the right panel. If this checkbox is not checked, the parameter estimations will be visualized in the right panel, after the estimation framework finishes running.
- Export Name: This field defines the filename of the exported temperature estimation animation file (in text). By default, the filename is the temperature data filename prepended by 'animation' and the file type is '.avi'.

- **Export Directory:** This field defines where to save the temperature estimation evolution animation file (in text). We can either type it in or click 'Browse', then locate and select the directory in the pop-up window. The default directory is where the observed temperature data file is located at.
- **Export Video:** By checking this checkbox, the animated temperature estimation evolution will be exported to a video file. The filename of the video is defined in the 'Export File' field and the video will be located at the directory, also defined in the 'Exported Directory'. Checking the 'Enable Real-time Animation' checkbox is required for exporting.
- **Start:** By clicking this button, the estimation framework will start.
- **Estimation Filename:** This is a field that defines the filename to save the sequential estimations of the parameters and temperature states (in text).
- **Save .mat:** By clicking this button, the estimations of the parameters and temperature states will be saved to the file, whose name is defined by the 'Estimation Filename' field.

6 Conclusions and Future Work

During this project, we studied in-office laryngeal laser procedures and the physical phenomena that occur during laser-tissue interactions. We also reviewed the challenges surgeons face when deciding whether the target tissue has been sufficiently treated. Our objective was to address these challenges through the identification of tissue types by studying their unique thermal responses to a surgical laser.

To study the thermal responses, we chose the Gaussian function to model the instantaneous temperature observed on the tissue surface. With this model, amplitude A and standard deviation σ are the two time-varying parameters that define the thermal response, so we needed to estimate and study their behaviors. We built an estimation framework using the Bayesian framework and the ensemble Kalman Filter (EnKF) to estimate the evolution of A and σ . We conducted observational tests under different framework settings to observe the efficiency of the estimation framework with various sets of simulated temperature data, including single pulse and multiple pulses temperature data (with residual heat and with center location changes).

These observational tests allowed us to study the effects of each input to the EnKF estimation framework on the estimation results in two aspects: the accuracy of estimation and the time efficiency. We observed the accuracy by comparing the estimated parameter values to the modeled values (through studying the absolute errors) and we observed the time efficiency by measuring the execution time. We observed that there are four EnKF settings that only affect the accuracy of the resulting estimations: the prior distribution type, the observational error covariance D , the model prediction error C , and the random walk deviation. We also observed that there are two other EnKF settings that affect both the accuracy and time efficiency: the ensemble size and the grid size in pre-processing observed temperature data. We tested different types of distributions for the initial prior distributions, and we observed that Uniform distributions, which are symmetric, gave us better estimations than other skewed distributions such as Exponential distributions and Poisson distributions. We tested different observational error covariances (D) and model prediction errors (C). We observed that for a set of accurate observed temperature data, smaller δ_D^2 (below 0.01) and smaller δ_C^2 (below 0.05) can provide better performance. We also observed different random walk deviations and we concluded that for parameters that vary faster (such as A and σ) would require a larger standard deviation. For parameters that vary slower (such as μ when we assume μ is slightly changing over time), a smaller standard deviation results in better performance. For our data simulation, we observed that defining 0.01 as the random walk standard deviation for A and σ and 0.001 for μ gave a reliable performance. Furthermore, an ensemble size of 500 and 20 as the grid size in data pre-processing can ensure good performance and reasonable execution time.

With the estimated temperature parameter behaviors, we developed machine learning models that take the parameter estimations and determine the different types of tissue. Two linear classification models, LDA and SVM were selected. These models were proven to be effective in correctly distinguishing the tissue type based on features derived from the estimated amplitude A and standard deviation σ sequences. The four features for classification were the maximum of the estimated amplitude sequence A_{max} , the mean of the estimated A_μ , the maximum of the standard deviation sequence σ_{max} , and the mean of the standard deviation sequence σ_μ . These four features proved effective for the classification of two tissue types, generated with amplitude functions differing in their peak temperature. The classification was able to effectively distinguish a difference of 30 °C between simulated tissue types.

Classifying the thermal response of laser-tissue interaction can be expanded significantly. This work covers just four simple features for classification based on the mean and maximum of estimated sequences. There are likely more possible features that will allow for more robust and effective classification of thermal responses of tissue. Additionally, further work can be done to test nonlinear machine learning models or machine learning models that accept sequences. Eventually, the goal is to create a classification model that allows for real-time classification based on a live feed of thermal data from a camera.

In this paper, we covered the accomplishment to quantify the thermal response of a laser-tissue interaction in a mathematical sense. However, additional work has to be done to transfer this information into a format

that can be used by a medical professional in a surgical application. The estimation of amplitude and standard deviation sequences should be tied back to known parameters of the tissue such as water content or whether tissue is benign or malignant.

Before our approach can be applied to in-office procedures, we still need to verify the estimation framework and classification models with experimental temperature data. For the verification, we built an experimental setup to collect thermal data. The setup successfully integrated a computer control system to fire the laser and collect data using a thermal camera. However, due to the unexpected circumstances of the COVID-19 pandemic, we were unable to collect reliable data to achieve our goal of verifying the mathematical work done. The experimental setup created can be further improved with rigorous documentation of setup configurations and control of all experimental variables to generate repeatable data. The experimental side of this project can be expanded further by using biological tissues, such as chicken muscle as opposed to tissue phantom to more closely represent an actual application of this project.

References

- [1] J. Anderson, Y. Bensoussan, R. Townsley, and E. Kell, “In-Office Endoscopic Laryngeal Laser Procedures: A Patient Safety Initiative,” *Otolaryngology–Head and Neck Surgery*, vol. 159, no. 1, pp. 136–142, 2018, pMID: 29737927. [Online]. Available: <https://doi.org/10.1177/0194599818774511>
- [2] A. G. Del Signore, R. N. Shah, N. Gupta, K. W. Altman, and P. Woo, “Complications and Failures of Office-Based Endoscopic Angiolytic Laser Surgery Treatment,” *Journal of Voice*, vol. 30, no. 6, pp. 744 – 750, 2016. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0892199715002003>
- [3] H.-C. Hu, S.-Y. Lin, and Y.-T. Hung, “Feasibility and Associated Limitations of Office-Based Laryngeal Surgery Using Carbon Dioxide Lasers,” *JAMA Otolaryngology–Head and Neck Surgery*, vol. 143, no. 5, pp. 485–491, 05 2017. [Online]. Available: <https://doi.org/10.1001/jamaoto.2016.4129>
- [4] M. Kumar, R. Chawla, and M. Goyal, “Topical anesthesia,” *Journal of Anaesthesiology Clinical Pharmacology*, vol. 31, no. 4, pp. 450–456, 2015. [Online]. Available: <http://www.joacp.org/article.asp?issn=0970-9185;year=2015;volume=31;issue=4;spage=450;epage=456;aulast=Kumar;t=6>
- [5] K. M. Tibbetts and C. B. Simpson, “Office-Based 532-Nanometer Pulsed Potassium-Titanyl-Phosphate Laser Procedures in Laryngology,” *Otolaryngologic Clinics of North America*, vol. 52, no. 3, p. 537–557, 2019. [Online]. Available: [https://www.oto.theclinics.com/article/S0030-6665\(19\)30013-1/](https://www.oto.theclinics.com/article/S0030-6665(19)30013-1/)
- [6] S. M. Zeitels, L. M. Akst, J. A. Burns, R. E. Hillman, M. S. Broadhurst, and R. R. Anderson, “Office-Based 532-nm Pulsed KTP Laser Treatment of Glottal Papillomatosis and Dysplasia,” *Annals of Otology, Rhinology & Laryngology*, vol. 115, no. 9, 2006. [Online]. Available: <https://doi.org/10.1177/000348940611500905>
- [7] (2018, Jul) Laser surgery through the mouth. [Online]. Available: www.cancerresearchuk.org/about-cancer/laryngeal-cancer/treatment/surgery/laser-surgery
- [8] M. Rubinstein and W. Armstrong, “Transoral laser microsurgery for laryngeal cancer: a primer and review of laser dosimetry,” *Lasers Med Sci*, vol. 26, no. 1, pp. 113–124, 2011, pMID: 20835840. [Online]. Available: <https://pubmed.ncbi.nlm.nih.gov/20835840>
- [9] S. L. Halum and A. C. Moberly, “Patient Tolerance of the Flexible CO₂ Laser for Office-based Laryngeal Surgery,” *Journal of Voice*, vol. 24, no. 6, pp. 750 – 754, 2010. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0892199709000629>
- [10] A. M. Store. Olympus ENF-V2 Flexible Video Rhino Laryngoscope. [Online]. Available: <https://aamedicalstore.com/products/olympus-enf-v2-flexible-video-rhino-laryngoscope>
- [11] L. Fichera, *Cognitive Supervision for Robot-Assisted Minimally Invasive Laser Surgery*, ser. Springer theses. SpringerOpen, 2016.
- [12] M. H. Niemz, *Laser-Tissue Interactions: Fundamentals and Applications*, 4th ed. Springer International Publishing, 2019.
- [13] D. Pardo, L. Fichera, D. Caldwell, and L. Mattos, “Learning Temperature Dynamics on Agar-Based Phantom Tissue Surface During Single Point CO₂ Laser Exposure,” *Neural Processing Letters*, vol. 42, no. 1, pp. 55,70, Aug. 2015.
- [14] A. Arnold and A. L. Lloyd, “An approach to periodic, time-varying parameter estimation using nonlinear filtering,” *Inverse Problems*, vol. 34, 2018. [Online]. Available: <https://iopscience.iop.org/article/10.1088/1361-6420/aad3e0/pdf>
- [15] K. Campbell, L. Staugler, and A. Arnold, “Estimating Time-Varying Applied Current in the Hodgkin-Huxley Model,” *Applied Sciences*, vol. 10, no. 2, 2020. [Online]. Available: <https://www.mdpi.com/2076-3417/10/2/550>

- [16] S. Rastegar, M. van Gemert, A. Welch, and L. Hayes, "Laser ablation of discs of agar gel," *National Center for Biotechnology Information*, vol. 33, no. 1, pp. 133–140, 1988. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pubmed/3353447>
- [17] L. Fichera, D. Pardo, P. Illiano, D. G. Caldwell, and L. S. Mattos, "Feed forward incision control for laser microsurgery of soft tissue," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 1235–1240.
- [18] A. L. McKenzie, "Physics of thermal processes in laser-tissue interaction," *Physics in Medicine and Biology*, vol. 35, no. 9, pp. 1175–1210, sep 1990. [Online]. Available: <https://doi.org/10.1088>
- [19] A. Welch, "The thermal response of laser irradiated tissue," *IEEE Journal of Quantum Electronics*, vol. 20, no. 12, pp. 1471–1481, December 1984.
- [20] "PANDA Robotic Arm." [Online]. Available: <https://www.generationrobots.com/en/403317-panda-robotic-arm.html>
- [21] "Aura XP Laser - ENT," ForTec Medical. [Online]. Available: <https://www.fortecmedical.com/fortec-product/aura-xp-laser-ent/>
- [22] M. Wolbarsht, "Laser surgery: CO₂ or HF," *IEEE Journal of Quantum Electronics*, vol. 20, no. 12, pp. 1427–1432, 1984.
- [23] S. Russo, P. Dario, and A. Menciassi, "A Novel Robotic Platform for Laser-Assisted Transurethral Surgery of the Prostate," *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 2, pp. 489–500, Feb 2015.
- [24] "FLIR A655sc," FLIR, Apr 2018. [Online]. Available: <https://flir.netx.net/file/asset/10275/original>
- [25] N. Hagen and E. L. Dereniak, "Gaussian profile estimation in two dimensions," *Applied Optics*, vol. 47, no. 36, pp. 6842–6845, 2008.
- [26] R. Yoho and D. O'Neil, "Duration of General Anesthesia and Surgical Outcome," 2006. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.501.7332&rep=rep1&type=pdf>
- [27] S. Gillijns, O. Mendoza, J. Chandrasekar, B. De Moor, D. Bernstein, and A. Ridley, "What is the ensemble Kalman filter and how well does it work?" in *2006 American Control Conference*. IEEE, 2006, p. 6 pp.
- [28] R. A. Fisher, "The Use Of Multiple Measurements In Taxonomic Problems," *Annals of Eugenics*, vol. 7, no. 2, p. 179–188, 1936. [Online]. Available: <https://digital.library.adelaide.edu.au/dspace/bitstream/2440/15227/1/138.pdf>
- [29] W. R. Klecka, *Discriminant analysis*, ser. A Sage university paper. Quantitative applications in the social sciences ; 19. Beverly Hills, Calif: Sage Publications, 1980.
- [30] C. Cortes and V. Vapnik, "Support-Vector Networks," *Machine Learning*, vol. 20, p. 273–297, 1995. [Online]. Available: http://image.diku.dk/imagecanon/material/cortes_vapnik95.pdf
- [31] K. Ross and R. Gordon, "Water in malignant tissue, measured by cell refractometry and nuclear magnetic resonance," 1982. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pubmed/7143438>
- [32] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer New York, 2016. [Online]. Available: <http://users.isr.ist.utl.pt/~wurmd/Livros/school/Bishop-PatternRecognitionAndMachineLearning-Springer2006.pdf>

Appendix A Mass-Spring System: An Application of EnKF

Before applying the ensemble Kalman filter (EnKF) to estimate parameters of temperature data, we first apply this technique to a simpler model: the mass-spring system ODE model. To practice the application of the EnKF, we are following the three steps below:

- Step 1: Apply the traditional Kalman filter to estimate the model states of the spring system.
- Step 2: Use the EnKF to estimate the model states only.
- Step 3: Apply the EnKF for both model state and parameter estimations.

These steps are prompted to prepare us in building the time-varying parameter estimation framework for a laser-tissue interaction. The ordinary differential equation (ODE) that describes the damped mass-spring oscillator is as follows:

$$m \cdot p''(t) + b \cdot p'(t) + k \cdot p(t) = 0 \quad (31)$$

where $p(t)$ is the position of the oscillator at time t , $p'(t)$, is the velocity of the oscillator at time t , $p''(t)$ is the acceleration of the oscillator at time t , m is the mass, b is the damping coefficient, and k is the spring constant. Before estimating the states and parameters of the spring system, it is essential to define them. The models states are position and velocity. Meanwhile, the parameters are the damping coefficient b , spring constant k , and mass of the oscillator m . In Step 1 & 2 we assume b , k , m to be known parameters and in Step 3 we assume b and k to be unknown and m still be a known parameter.

A.1 State Estimation with Traditional Kalman Filter

The goal is to estimate the states of the spring system model by applying the traditional Kalman filter. Firstly, we use v to represent the velocity $p'(t)$ and a to represent the acceleration $p''(t)$. Then we can rewrite the ODE to solve for the acceleration of the oscillator, $p''(t)$:

$$p''(t) = (-k/m) \cdot p(t) + (-b/m) \cdot p'(t) \quad (32)$$

$$a = (-k/m) \cdot v + (-b/m) \cdot p \quad (33)$$

As we are interested in estimating the states only, we define the initial state using a vector x that contains p and v , denoted as $x = \begin{bmatrix} p \\ v \end{bmatrix}$. From the vector x and taking the derivative, we can obtain the following system of ODEs:

$$\frac{dx}{dt} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{b}{m} \end{bmatrix} x = Ax \quad (34)$$

As the ODE is continuous and the Kalman filter is limited to discrete models, we can discretize the ODE by finding its state evolution equation and observation equation. These two equations are helpful when applying the filter, as they help predict and update the predictions. The state evolution equation is:

$$x_{j+1} = (I + hA)x_j + V_{j+1} = Fx_j + V_{j+1}, \quad F = I + hA \quad (35)$$

where j is the discrete time, x_{j+1} is the predicted state vector at time $j + 1$, I is the identity matrix, h is the change of time, x_j is the state vector at time j , and V_{j+1} is the error at time $j + 1$. We then define F as the state transition matrix, which can assist us when we apply the Kalman filter. The observation equation is:

$$y_j = \begin{bmatrix} 1 & 0 \end{bmatrix} x_j + W_{j+1} \quad (36)$$

where y_j is the observation at time j , $\begin{bmatrix} 1 & 0 \end{bmatrix}$ is the observation matrix, x_j is the predicted state vector at time j , and W_{j+1} is the error at time $j + 1$. Next, we define the initial states, $\bar{x}_{0|0}$, and state variance, $\Gamma_{0|0}$. For

traditional Kalman Filter, we need to choose the initial means and covariance. We choose the initial mean of position, $\bar{p}_{0|0}$ to be 1 and the initial mean of velocity, $\bar{v}_{0|0}$ to be 0. Therefore, we can initialize the initial states $\bar{x}_{0|0}$ as follows:

$$\bar{x}_{0|0} = \begin{bmatrix} \bar{p}_{0|0} \\ \bar{v}_{0|0} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (37)$$

For the initial covariance, $\Gamma_{0|0}$ is defined as a diagonal matrix multiplied by a constant 0.1:

$$\Gamma_{0|0} = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix} \quad (38)$$

In each iteration at time index j , we need to follow two steps: the prediction step and the observation update step. In the prediction step, we need to predict the mean of states at time index $j + 1$, $\bar{x}_{j+1|j}$, and the covariance at time index $j + 1$, $\Gamma_{j+1|j}$, without knowledge of the observation. To predict the mean of states $\bar{x}_{j+1|j}$, we use the state evolution equation, as follows:

$$\bar{x}_{j+1|j} = F\bar{x}_{j|j} \quad (39)$$

To predict $\Gamma_{j+1|j}$, we use the state evolution equation, as follows:

$$\Gamma_{j+1|j} = F\Gamma_{j|j}F^T + C \quad (40)$$

in which $C \in \mathbb{R}_{2 \times 2}$ is the system error that accounts for the noise in prediction. In the observation update step, we take in the observed states y_{j+1} and update our predictions. First, we compute the Kalman gain K_{j+1} :

$$K_{j+1} = \Gamma_{j+1|j}G^T(G\Gamma_{j+1|j}G^T + D)^{-1} \in \mathbb{R}_{2 \times 1} \quad (41)$$

in which $D \in \mathbb{R}_{1 \times 1}$ is the observation noise. Then we use the observation equation to compute the posterior mean and the error covariance:

$$\bar{x}_{j+1|j+1} = \bar{x}_{j+1|j} + K_{j+1}(y_{obs,j+1} - G\bar{x}_{j+1|j}), \quad \Gamma_{j+1|j+1} = \Gamma_{j+1|j} - K_{j+1}G\Gamma_{j+1|j} \quad (42)$$

where K_{j+1} is the Kalman gain, $y_{obs,j+1}$ is the observed position value at time index $j + 1$, and $\bar{x}_{j+1|j+1}$ is the posterior mean at time index $j + 1$. Assuming we get our last observation at time index M , we need to repeat this prediction-correction process for each j from 1 to the last time index, and we will get a sequence of posterior mean \bar{x} , which is plotted in Figure 40:

$$\bar{x} = [\bar{x}_{1|1}, \bar{x}_{2|2}, \dots, \bar{x}_{M|M}] = [(\bar{p}_{1|1}, \bar{v}_{1|1}), (\bar{p}_{2|2}, \bar{v}_{2|2}), \dots, (\bar{p}_{M|M}, \bar{v}_{M|M})] \quad (43)$$

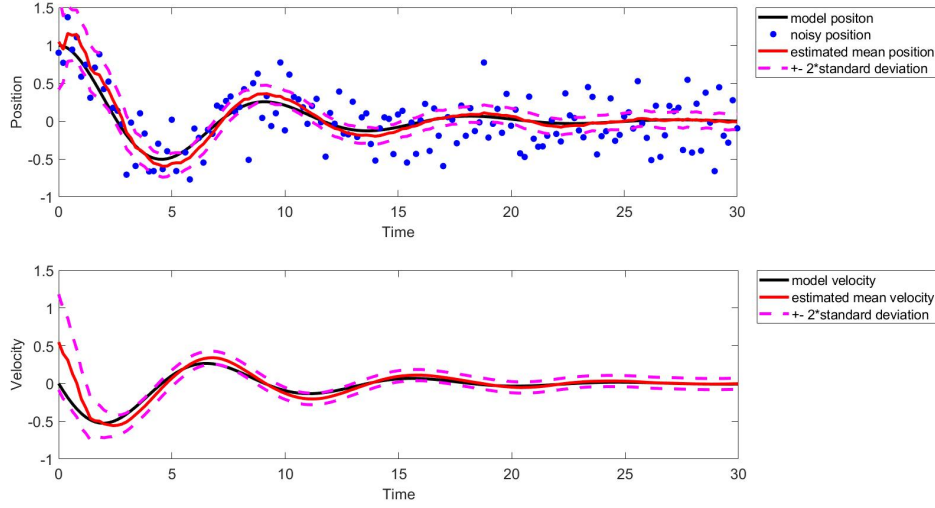


Figure 40: States (position p and velocity v) estimations resulting from the estimation framework (which is built from applying the traditional Kalman Filter). These plots display the model states (black), posterior mean values (red), ± 2 standard deviation (pink), and observed states (blue).

A.2 State Estimation with EnKF

Now, our goal is to estimate the states of the spring system model with ensemble Kalman filter (EnKF). We now treat p and v as random variables with probability distributions, and we want to compute the posterior distribution from the prior distribution first and then compute the posterior mean. Let the ensemble size be $N_{ens} = 100$, the ensemble at time index j is $S_{j|j}$ and it has the following form:

$$S_{j|j} = \{x_{j|j}^1, x_{j|j}^2, \dots, x_{j|j}^{N_{ens}}\} \quad (44)$$

The initial prior ensemble consists of random draws from the prior distributions. Here, we continue to let the initial prior distributions, $p_{0|0}$, be $\mathcal{N}(0, 0.2^2)$ that the prior mean value of $p_{0|0}$ is 0 and the standard deviation is 0.2. The initial prior distribution, $v_{0|0}$, is $\mathcal{N}(0, 0.2^2)$, that the prior mean of $v_{0|0}$ is 0 and the standard deviation is 0.2. The EnKF method contains the same prediction step and observation update step as the traditional Kalman Filter. To discretize the system, we use the same state evolution equation and observation equation, as described in Step 1. In the prediction step, for each observation sample, we predict the states, $x_{j+1|j}^n$ with the state evolution matrix from the state evolution equation, as follows:

$$x_{j+1|j}^n = (I + hA)x_{j|j}^n + C = Fx_{j|j}^n + C, \quad n = 1, 2, \dots, N_{ens} \quad (45)$$

in which $C \in \mathbb{R}_{2 \times 1}$ is the system error that accounts for the noise in prediction. Then, we predict the position value, $y_{j+1|j}^n$ with the observation equation:

$$y_{j+1|j}^n = [1 \quad 0] x_{j+1|j}^n, \quad n = 1, 2, \dots, N_{ens} \quad (46)$$

Then, we compute the mean of $x_{j+1|j}^n$ and $y_{j+1|j}^n$ as follows:

$$\bar{x}_{j+1|j} = \frac{1}{N_{ens}} \sum_{n=1}^{N_{ens}} x_{j+1|j}^n, \quad \bar{y}_{j+1|j} = \frac{1}{N_{ens}} \sum_{n=1}^{N_{ens}} y_{j+1|j}^n \quad (47)$$

After computing the predicted means of the states and the position, we want to compute the difference between the predicted states and their means, as well as for the predicted observations:

$$E_x = x_{j+1|j}^n - \bar{x}_{j+1|j}, \quad E_y = y_{j+1|j}^n - \bar{y}_{j+1|j} \quad (48)$$

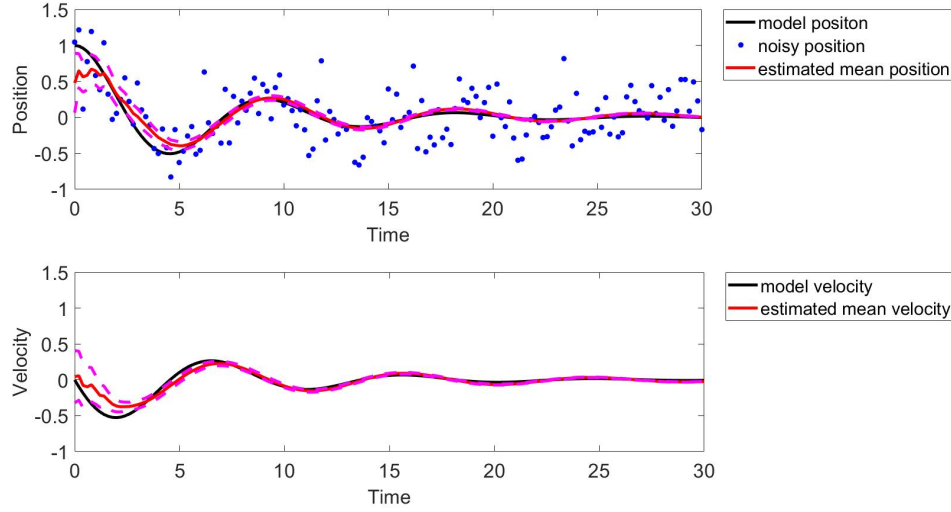


Figure 41: States (position p and velocity v) estimations resulting from the estimation framework (which is built from applying the EnKF). These plots display the model states (black), estimated states (red), ± 2 standard deviation (pink), and observed states (blue).

With the predicted position $y_{j+1|j}^n$, we correct our predictions by taking the observed position value $y_{obs,j+1}$, into account. We compute the Kalman gain, K_{j+1} , first, using the following equation:

$$K_{j+1} = \left(\frac{E_x E_y^T}{N_{ens} - 1} \right) \left(\frac{E_y E_y^T}{N_{ens} - 1} + D \right)^{-1} \in \mathbb{R}_{2 \times 1} \quad (49)$$

in which $D \in \mathbb{R}_{1 \times 1}$ is the observation noise. With the Kalman gain, we can compute the posterior ensemble, $x_{j+1|j+1}^n$ at time index $j + 1$:

$$x_{j+1|j+1}^n = x_{j+1|j}^n + K_{j+1}(y_{obs,j+1} - y_{j+1|j}^n), \quad n = 1, 2, \dots, N_{ens} \quad (50)$$

in which $y_{obs,j+1}$ is the observed position at time index $j + 1$. Lastly we compute the mean value of the posterior distribution $x_{j+1|j+1}^n$ at time index $j + 1$ as $\bar{x}_{j+1|j+1}$ and save the posterior mean in a sequence. Assuming we get our last observation at time index M , we repeat the prediction-correction procedure for each j from 1 to M . Figure 41 displays the observed states and the estimated states, p and v , resulting from the EnKF.

A.3 State and Parameter Estimation with EnKF

In Steps 1 and 2, we assumed that the damping coefficient b , spring constant k , and the mass m are known parameters. Now, we want to treat b and k as unknown parameters and use the EnKF to estimate the two states, position p and velocity v , and the two unknown parameters, b and k . Notice that we are still treating m as a known parameter. We defined the states and parameters using a vector z that contains p , v , b , and k values, at time j , denoted as:

$$z_{j|j} = \begin{bmatrix} p_{j|j} \\ v_{j|j} \\ b_{j|j} \\ k_{j|j} \end{bmatrix} \quad (51)$$

Let the ensemble size be N_{ens} and our ensemble $S_{j|j}$ at time index j has the following form:

$$S_{j|j} = \{z_{j|j}^1, z_{j|j}^2, \dots, z_{j|j}^{N_{ens}}\} \quad (52)$$

The initial prior ensemble consists of random draws from the initial prior distributions of each state or parameter. Here, we continue to let the initial prior distributions, $p_{0|0}$, be $\mathcal{N}(0, 0.2^2)$ such that the prior mean value of $p_{0|0}$ is 0 and the standard deviation is 0.2. We let the initial prior distribution, $v_{0|0}$, be $\mathcal{N}(0, 0.2^2)$ in which the prior mean of $v_{0|0}$ is 0, and the standard deviation is 0.2. The initial prior distribution of the damping coefficient, $b_{0|0}$, is $\mathcal{N}(2, 5^2)$ such that the prior mean is 2 and the standard deviation is 5. The initial prior distribution of the spring constant, $k_{0|0}$, is $\mathcal{N}(2, 5^2)$ in which the prior mean is 2 and the standard deviation is 5.

Similarly, we have the prediction step and the observation update step of the EnKF. However, instead of defining A to be static and based on the known b , k , and m , we define the state evolution matrix, A_j^n , with $b_{j|j}^n$ and $k_{j|j}^n$ in $z_{j|j}^n$:

$$A_j^n = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k_{j|j}^n}{m} & -\frac{b_{j|j}^n}{m} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (53)$$

In the prediction step, we use the state transition matrix F from the state evolution equation to help us compute the prediction ensemble statistics, $z_{j+1|j}^n$:

$$z_{j+1|j}^n = (I_4 + hA_j^n)z_{j|j}^n + C = Fz_{j|j}^n + C, \quad n = 1, 2, \dots, N_{ens} \quad (54)$$

in which $C \in \mathbb{R}_{4 \times 1}$ is the system error that accounts for the noise in prediction. Like previous steps, we also need to use the observation matrix to predict the position state, $y_{j+1|j}^n$:

$$y_{j+1|j}^n = Gz_{j+1|j}^n = [1 \quad 0 \quad 0 \quad 0] z_{j+1|j}^n, \quad n = 1, 2, \dots, N_{ens} \quad (55)$$

Then, we compute the mean of $z_{j+1|j}^n$ and $y_{j+1|j}^n$ as follows:

$$\bar{z}_{j+1|j} = \frac{1}{N_{ens}} \sum_{n=1}^{N_{ens}} z_{j+1|j}^n, \quad \bar{y}_{j+1|j} = \frac{1}{N_{ens}} \sum_{n=1}^{N_{ens}} y_{j+1|j}^n \quad (56)$$

After computing the means, we want to compute the difference between the predicted states and their means, as well as for the predicted observations:

$$E_z = z_{j+1|j}^n - \bar{z}_{j+1|j}, \quad E_y = y_{j+1|j}^n - \bar{y}_{j+1|j} \quad (57)$$

With the predicted position values $y_{j+1|j}^n$, we can correct our predictions by taking the observed position value $y_{obs,j+1}$, into account. We compute the Kalman gain, K_{j+1} , first, using the following equation:

$$K_{j+1} = \left(\frac{E_z E_y^T}{N_{ens} - 1} \right) \left(\frac{E_y E_y^T}{N_{ens} - 1} + D \right)^{-1} \in \mathbb{R}_{4 \times 1} \quad (58)$$

in which $D \in \mathbb{R}_{1 \times 1}$ is the observation noise. With the Kalman gain, we can compute the posterior ensemble, $z_{j+1|j+1}^n$ at time index $j+1$:

$$z_{j+1|j+1}^n = z_{j+1|j}^n + K_{j+1}(y_{obs,j+1} - y_{j+1|j}^n), \quad n = 1, 2, \dots, N_{ens} \quad (59)$$

in which $y_{obs,j+1}$ is the observed position at time index $j+1$. Lastly we compute the mean value of the posterior distribution $z_{j+1|j+1}^n$ at time index $j+1$ as $\bar{z}_{j+1|j+1}$ and save the posterior mean in a sequence. Similarly, assuming we get our last observation at time index M , we repeat the prediction-correction procedure for each j from 1 to M . As we finish the estimation at time index M , we get a sequence of posterior mean:

$$\bar{z} = [\bar{z}_{1|1}, \bar{z}_{2|2}, \dots, \bar{z}_{M|M}] = [(p_{1|1}, v_{1|1}, b_{1|1}, k_{1|1}), (p_{2|2}, v_{2|2}, b_{2|2}, k_{2|2}), \dots, (p_{M|M}, v_{M|M}, b_{M|M}, k_{M|M})] \quad (60)$$

Figure 42 and Figure 43 displays the state (p and v) and parameter (b and k) estimations resulting from the EnKF estimation framework.

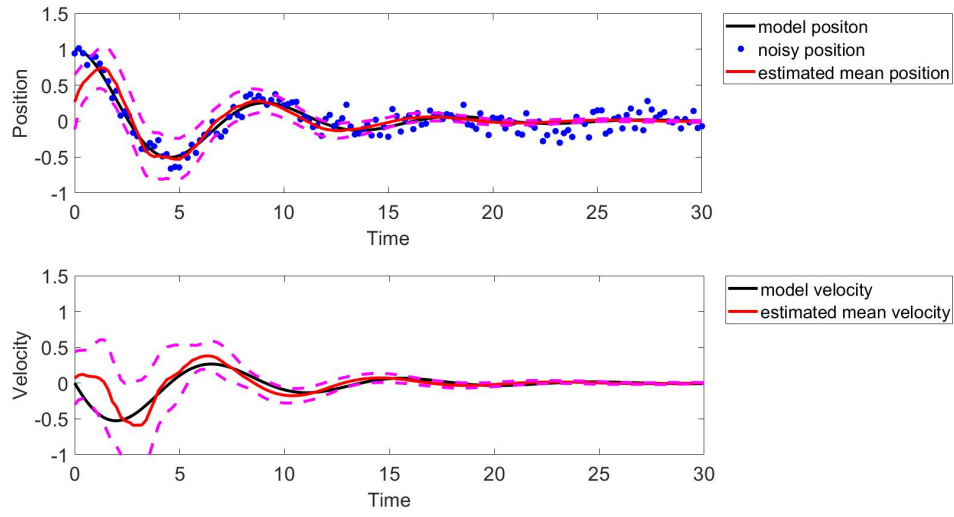


Figure 42: States (position p and velocity v) estimations resulting from the estimation framework built from applying the EnKF. These plots display the model states (black), estimated states (red), ± 2 standard deviation (pink), and observed states (blue).

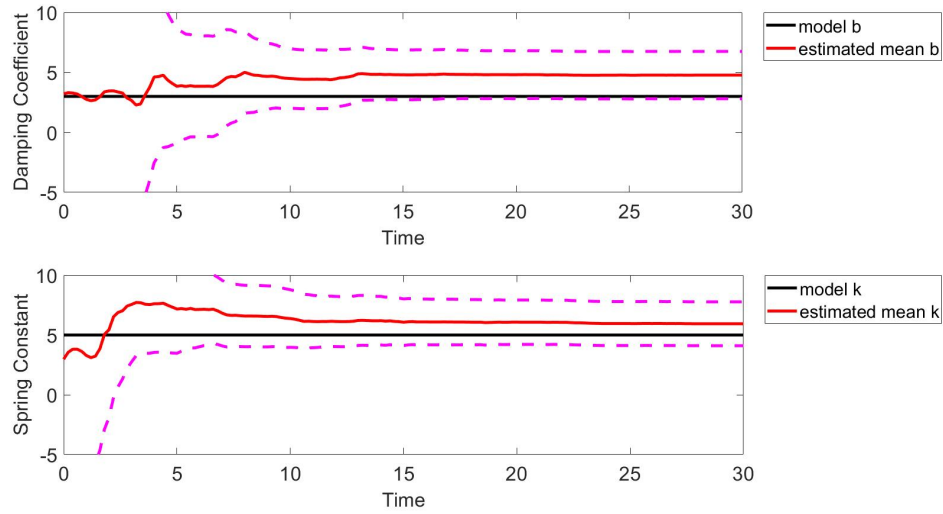


Figure 43: Parameter (damping coefficient b and spring constant k) estimations resulting from the estimation framework built from applying the EnKF. These plots display the model states (black), estimated states (red), and ± 2 standard deviation (pink).

Appendix B Results of Estimation Framework Settings

B.1 Distribution Types Testing

We are interested in studying how the values of a random sample of the amplitude (A) and standard deviation (σ) from the prescribed prior distribution are selected, which is affected by the type of distribution. In return, the results of the estimation framework are affected by the random sample of A and σ . We tested the Normal, Exponential, Weibull, Poisson, and Uniform distributions. The input parameters for the different types of distribution vary. Therefore, we used appropriate input parameters for the various distributions.

In general, values around the skewed peak of the curve are more likely to be selected for skewed distributions, like the Exponential, Weibull, and Poisson distributions. Even the symmetric Normal distribution is more likely to generate values around its peak. On the other hand, the Uniform distribution ensures that all values between the upper and lower endpoints are equally likely to be generated. This section displays the common results of the observational tests we ran.

According to Figure 44, values are more likely to be selected around the skewed peak of the curve (between 0 and 1). The resulting estimated parameters are oscillating rapidly in the 0.5 to 1 second of the simulation, then the estimations become more accurate to the model parameters. Meanwhile, as shown in Figure 45, values are more likely to be randomly selected around the skewed peak of the curve (between 0 and 3). These estimations are similar, in terms of accuracy, to the estimations resulted from the exponentially distributed initial parameters. Although values around the skewed peak of the curve (between 0 and 1) are more likely to be selected, values generated from a Poisson distribution are less scattered, as compared to the exponential and Weibull distributions. Referring to Figure 46, the resulting estimations take a longer duration (as compared to the exponential and Weibull distributions) until it generates accurate estimations.

As shown in Figure 47, more random values are generated around the center peak of the curve and are more scattered throughout the curve. The resulting estimations are similar to that of the exponential and Weibull distributions, with a wider ± 2 standard deviation within the first 0.5 seconds. Whereas, according to Figure 48, values are equally likely to be generated within the given upper and lower endpoints. We favor this feature of the uniform distribution because we hypothesize that equally likely selected random values of the initial parameters would result in more accurate estimations. The estimations are similar to those that resulted from a normal distribution, but the estimated parameters are closer to the model parameters in the first second.

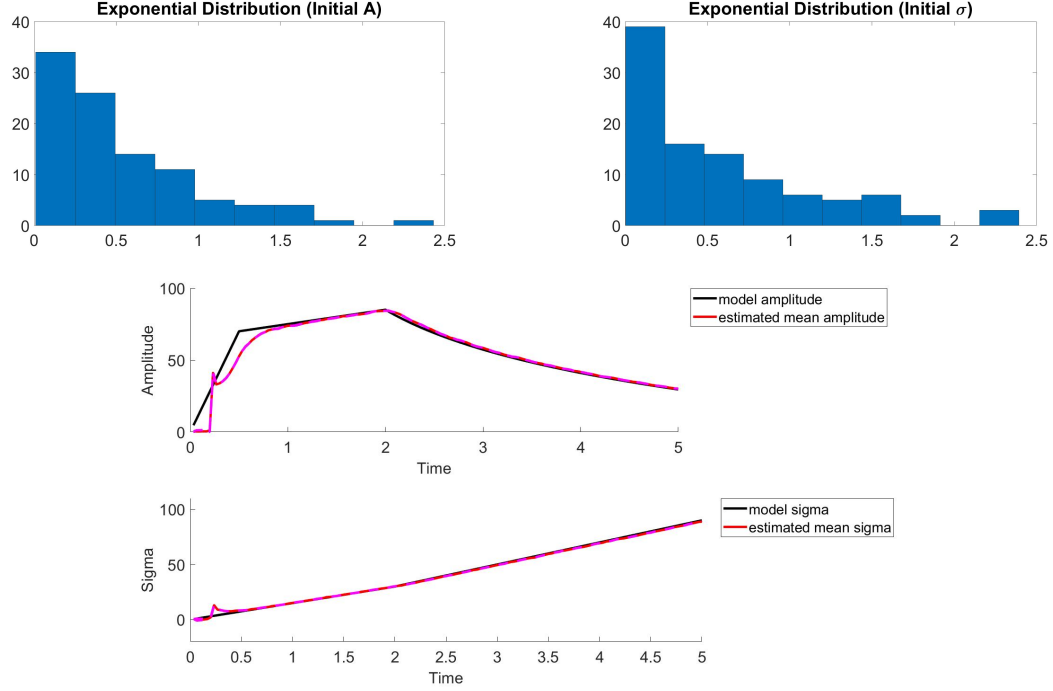


Figure 44: Exponentially distributed values for the initial parameters and the resulting parameter estimations. The estimation plots display the model parameter (in black), the estimated parameter (red), and the ± 2 standard deviation (pink).

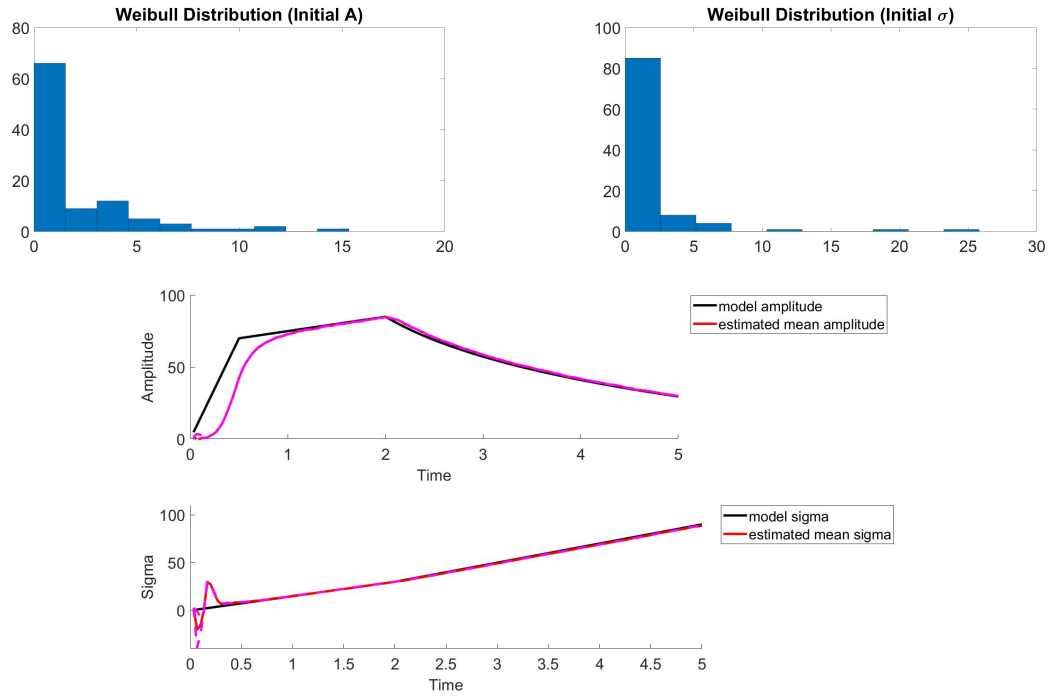


Figure 45: Random values from a Weibull distribution for the initial parameters and the resulting parameter estimations. The estimation plots display the model parameter (in black), the estimated parameter (red), and the ± 2 standard deviation (pink).

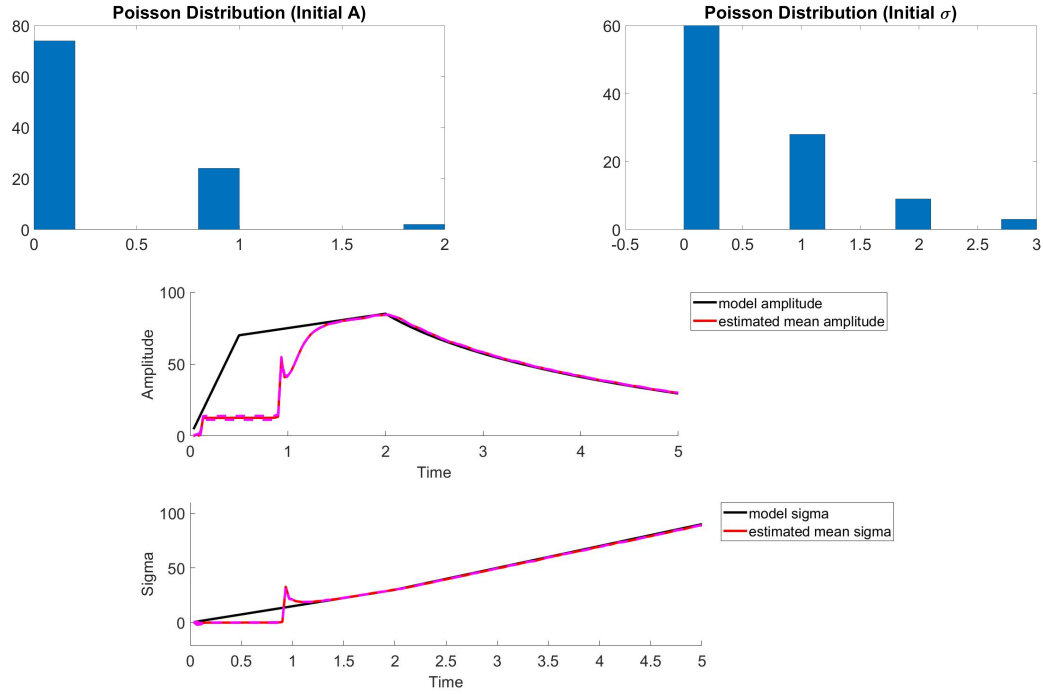


Figure 46: Random values from a Poisson distribution for the initial parameters and the resulting parameter estimations. The estimation plots display the model parameter (in black), the estimated parameter (red), and the ± 2 standard deviation (pink).

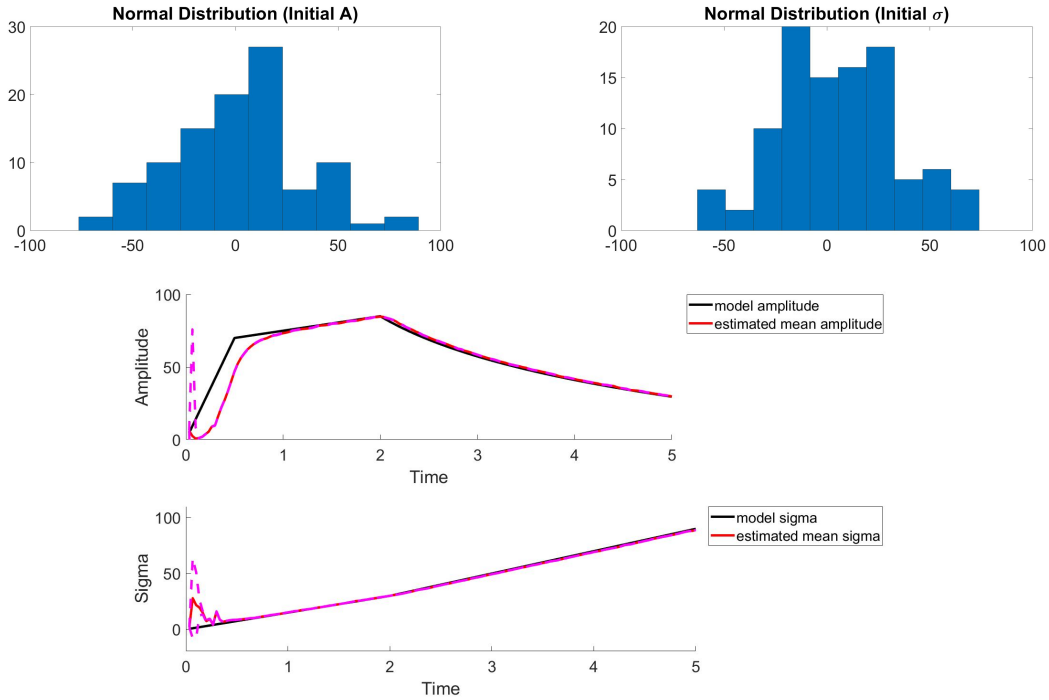


Figure 47: Normally distributed values for the initial parameters and the resulting parameter estimations. The estimation plots display the model parameter (in black), the estimated parameter (red), and the ± 2 standard deviation (pink).

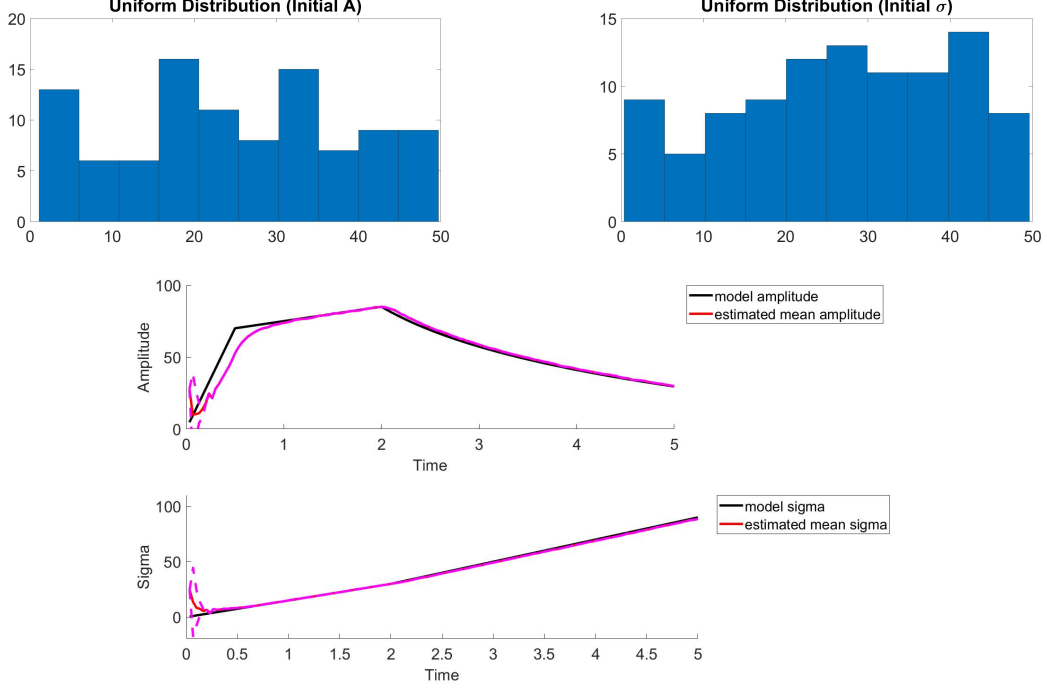


Figure 48: Uniformly distributed values for the initial parameters and the resulting parameter estimations. The estimation plots display the model parameter (in black), the estimated parameter (red), and the ± 2 standard deviation (pink).

B.2 Observation Error Testing

Our goal in this section is to determine a reasonable observation error (D) that provides accurate estimations and is realistic to laser-tissue experiments. Observation errors reflect how much the estimation framework trusts the observation points. We observed the effects of different D errors on the estimation framework with simulated data. To make convincing observations on various error values, we set the other input variables constant. The following are some of the primary inputs that were constant:

- Model Prediction Error (C)

$$C = \begin{bmatrix} 0.0001 & 0.0001 & \cdots & 0.0001 \\ 0.0001 & 0.0001 & \cdots & 0.0001 \\ \vdots & \vdots & \ddots & \vdots \\ 0.0001 & 0.0001 & \cdots & 0.0001 \end{bmatrix} \in \mathbb{R}_{20 \times 20} \quad (61)$$

- Initial A (Parameters of Uniform Distribution)
upper endpoint = 0, lower endpoint = 50
- Initial σ (Parameters of Uniform Distribution)
upper endpoint = 0, lower endpoint = 50
- Random walk, E (Parameters of Normal Distribution)
mean = 0, standard deviation = 0.01

Below are the common results from running tests on the effects of various observation errors (D), with different values of δ_D^2 (including $\delta_D^2 = 0.005, 0.01, 0.1, 0.5$, and 1), each test was run ten times:

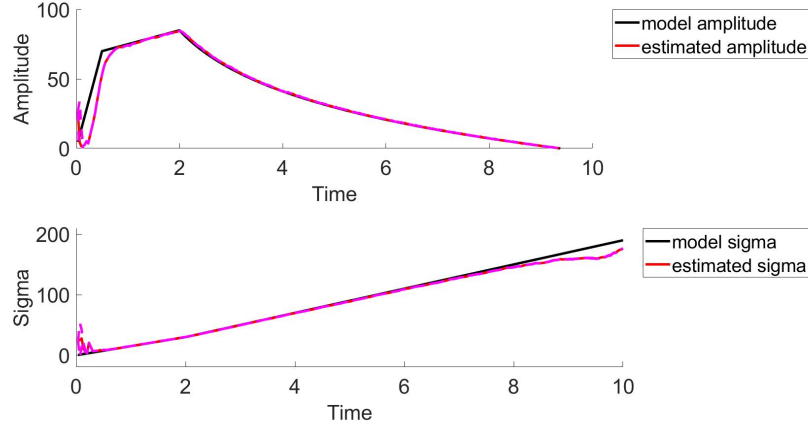


Figure 49: Resulting parameter estimation of a single pulse for testing various observation errors, when $\delta_D^2 = 0.005$. These plots display the model parameter (black), estimated parameter (red), and +/-2 standard deviation (pink).

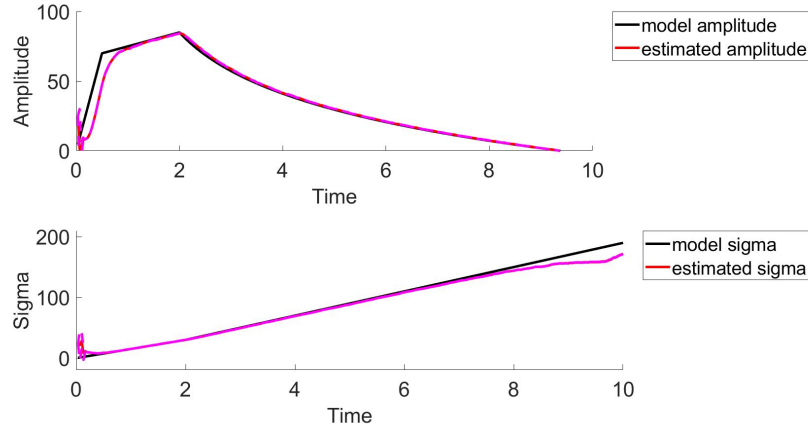


Figure 50: Common result of parameter estimation of a single pulse for testing various observation errors, when $\delta_D^2 = 0.01$. These plots display the model parameter (black), estimated parameter (red), and +/-2 standard deviation (pink).

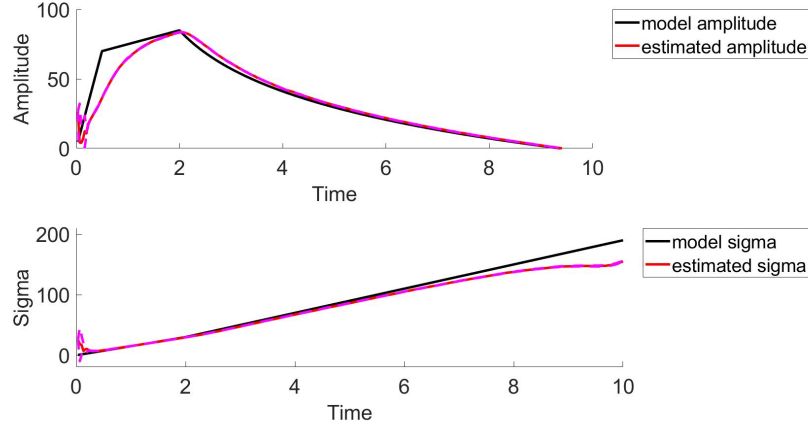


Figure 51: Common parameter estimations of a single pulse for testing various observation errors, when $\delta_D^2 = 0.1$. These plots display the model parameter (black), estimated parameter (red), and +/-2 standard deviation (pink).

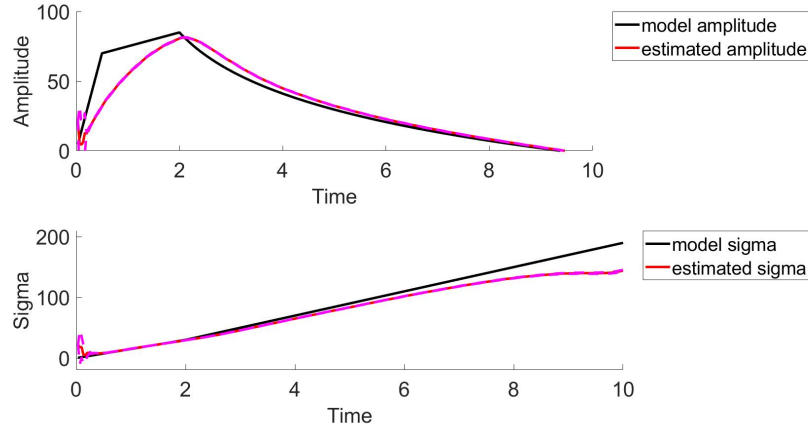


Figure 52: Common result of parameter estimation of a single pulse for testing various observation errors, when $\delta_D^2 = 0.5$. These plots display the model parameter (black), estimated parameter (red), and +/-2 standard deviation (pink).

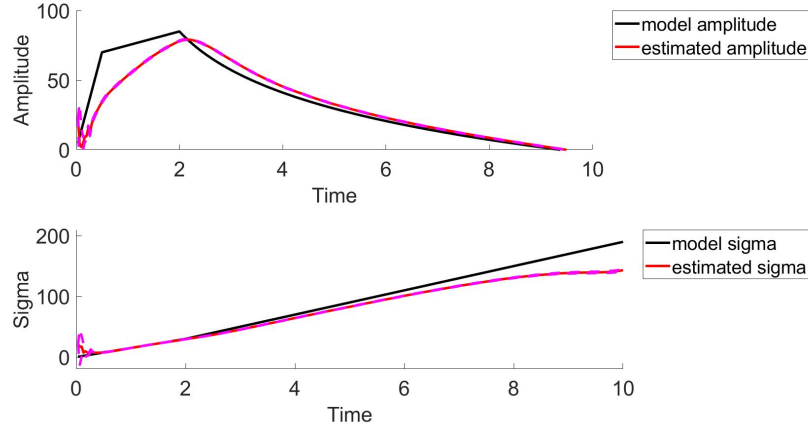


Figure 53: Common result of parameter estimation of a single pulse for testing various observation errors, when $\delta_D^2 = 1$. These plots display the model parameter (black), estimated parameter (red), and +/-2 standard deviation (pink).

B.3 Model Prediction Error Testing

We tested different values for the model prediction error, which reflects the degree to which the parameters are trusted. To observe the effect of changing the error value, we fixed the other input variables when we ran the tests. The following are some of the primary input variables that we set fixed in this observation:

- Observation Error (D)

$$D = \begin{bmatrix} 0.01 & 0 & \cdots & 0 \\ 0 & 0.01 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0.01 \end{bmatrix} \in \mathbb{R}_{400 \times 400} \quad (62)$$

- Initial A (Parameters of Uniform Distribution)
upper endpoint = 0, lower endpoint = 50
- Initial σ (Parameters of Uniform Distribution)
upper endpoint = 0, lower endpoint = 50
- Random walk, E (Parameters of Normal Distribution)
mean = 0, standard deviation = 0.01

We ran each test ten times to observe the common results. A relatively lower model prediction error would result in better performance. Below are the common results from running tests on the effects of various model prediction errors (C), with different values of δ_C^2 (including $\delta_C^2 = 0.0001, 0.001, 0.01, 0.05, 0.1, 0.5$, and 1):

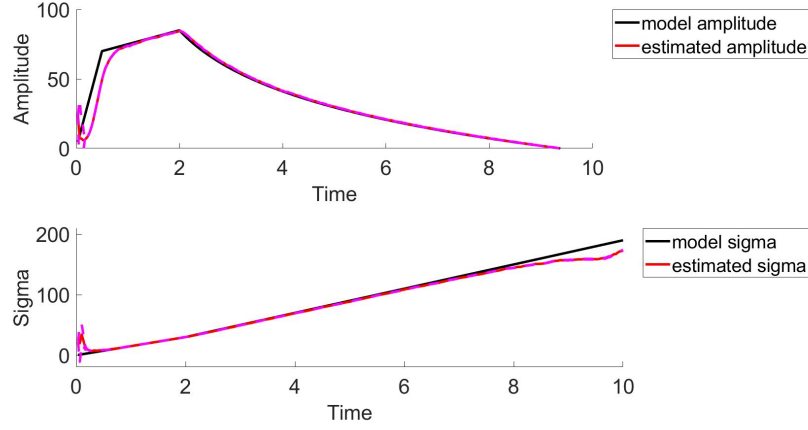


Figure 54: Common results from estimation framework on a single pulse for testing various model prediction errors, when $\delta_C^2 = 0.0001$. These plots display the model parameter (black), estimated parameter (red), and ± 2 standard deviation (pink).

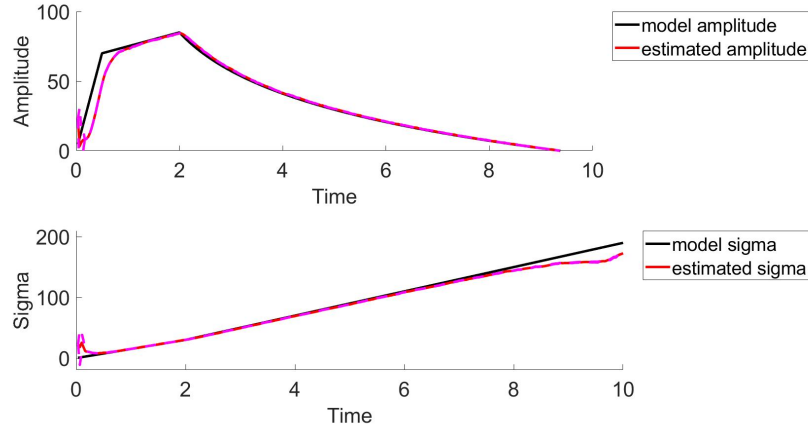


Figure 55: Common estimation results of a single pulse for testing various model prediction errors, when $\delta_C^2 = 0.001$. These plots display the model parameter (black), estimated parameter (red), and ± 2 standard deviation (pink).

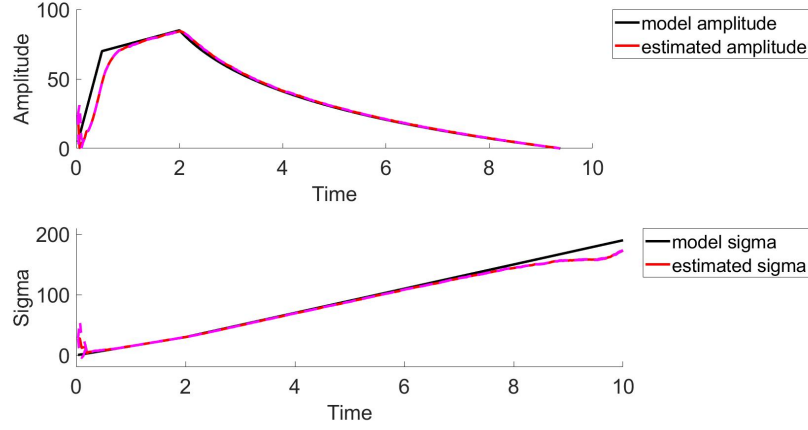


Figure 56: Parameter estimations of a single pulse for testing various model prediction errors, when $\delta_C^2 = 0.01$. These plots display the model parameter (black), estimated parameter (red), and ± 2 standard deviation (pink).

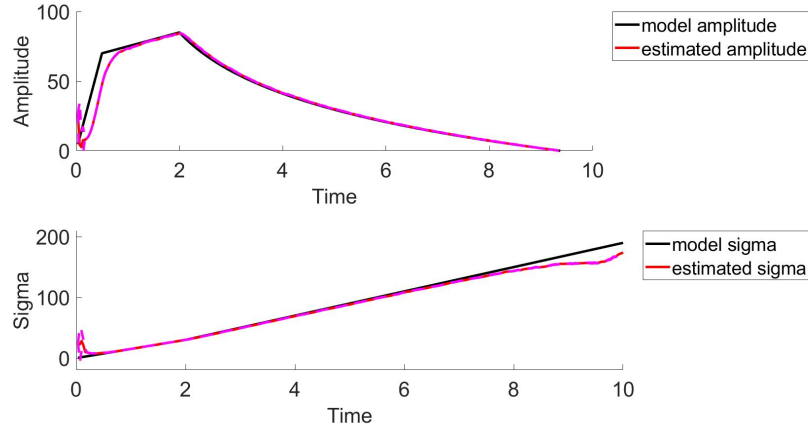


Figure 57: Parameter estimations of a single pulse for testing various model prediction errors, when $\delta_C^2 = 0.05$. These plots display the model parameter (black), estimated parameter (red), and ± 2 standard deviation (pink).

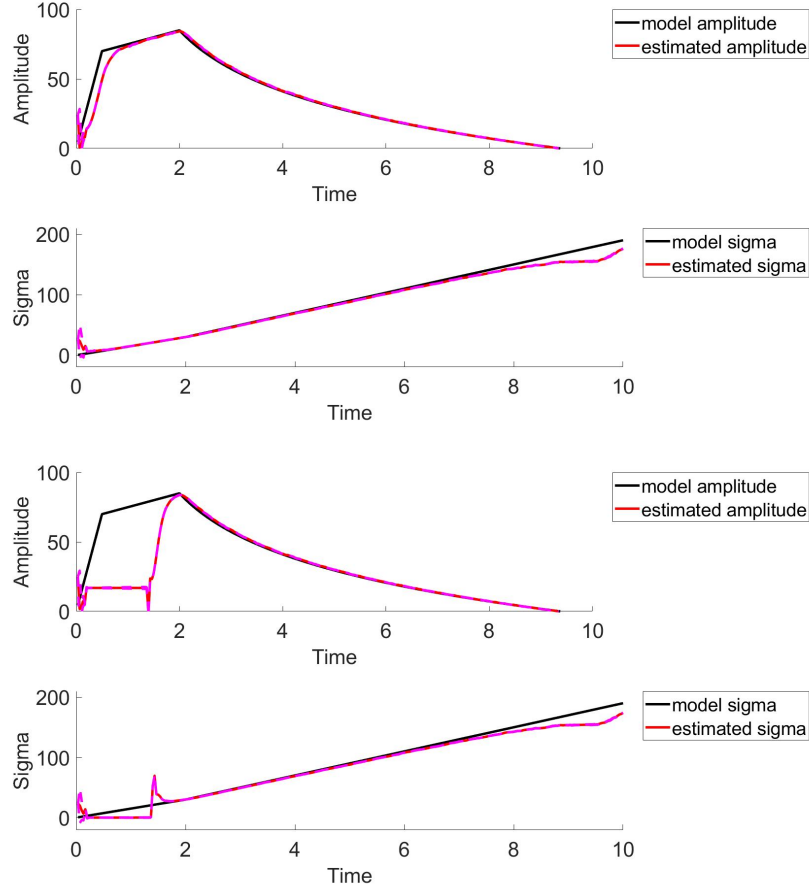


Figure 58: Common parameter estimation of a single pulse for testing various model prediction errors, when $\delta_C^2 = 0.1$. These plots display the model parameter (black), estimated parameter (red), and ± 2 standard deviation (pink). Some of the runs resulted in more accurate estimations (top), while other resulted in less accurate estimations (bottom).

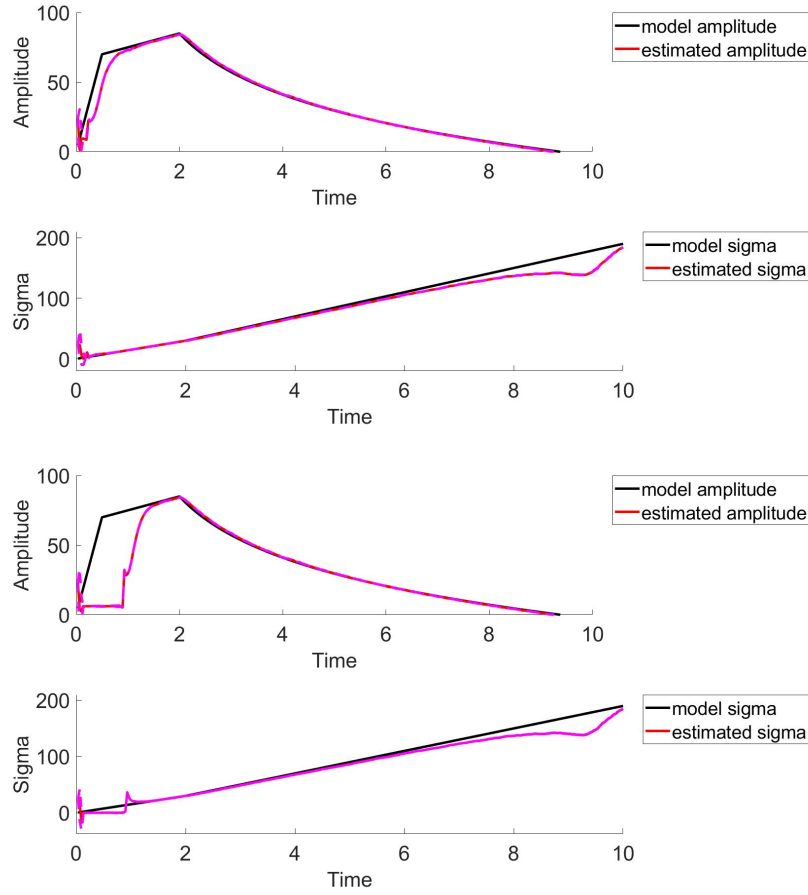


Figure 59: Common estimation results of a single pulse for testing various model prediction errors, when $\delta_C^2 = 0.5$. These plots display the model parameter (black), estimated parameter (red), and ± 2 standard deviation (pink). Some of the runs resulted in more accurate estimations (top), while other resulted in less accurate estimations (bottom).

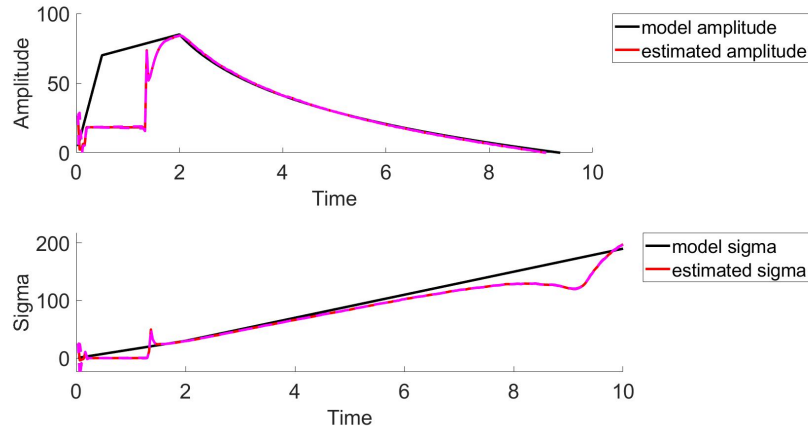


Figure 60: Common results from estimation framework on a single pulse for testing various Model Prediction errors, when $\delta_C^2 = 1$. These plots display the model parameter (black), estimated parameter (red), and ± 2 standard deviation (pink).

B.4 Random Walk Testing

B.4.1 Results from a Single Pulse Simulation

As the parameters, A and σ , are time-varying, we add a random walk, or a new random variable (E), to each parameter in the predict step of the EnFK. The E values are randomly selected values of a Normal distribution with a mean of 0 and a chosen standard deviation. We study the effects of different standard deviations on the accuracy of the results of the estimation framework. For testing purposes, we set the other input values of the framework as constants, and only change the value of the standard deviation of the random walks. The following are some of the primary inputs that we defined as constants for this observation:

- Observation Error (D)

$$D = \begin{bmatrix} 0.01 & 0 & \cdots & 0 \\ 0 & 0.01 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0.01 \end{bmatrix} \in \mathbb{R}_{400 \times 400} \quad (63)$$

- Model Prediction Error (C)

$$C = \begin{bmatrix} 0.0001 & 0.0001 & \cdots & 0.0001 \\ 0.0001 & 0.0001 & \cdots & 0.0001 \\ \vdots & \vdots & \ddots & \vdots \\ 0.0001 & 0.0001 & \cdots & 0.0001 \end{bmatrix} \in \mathbb{R}_{20 \times 20} \quad (64)$$

- Initial A (Parameters of Uniform Distribution)
upper endpoint = 0, lower endpoint = 50
- Initial σ (Parameters of Uniform Distribution)
upper endpoint = 0, lower endpoint = 50

The values of standard deviation that we tested includes 0.005, 0.01, 0.025, 0.05, and 0.1. To make observational conclusions, we tested each different standard deviation ten times. The results are shown in Figures 61, 62, 63, 64, and 65.

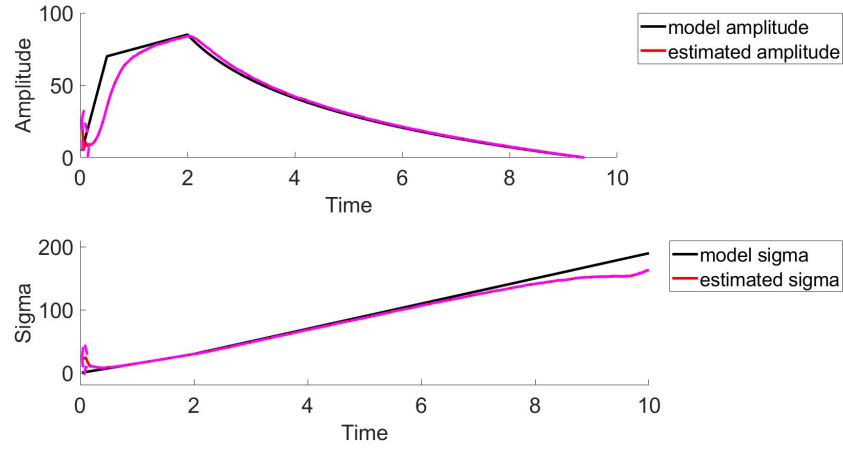


Figure 61: Estimation results for a single pulse with a standard deviation of 0.005 of random walks (A and σ). These plots display the model parameter (black), estimated parameter (red), and ± 2 standard deviation (pink).

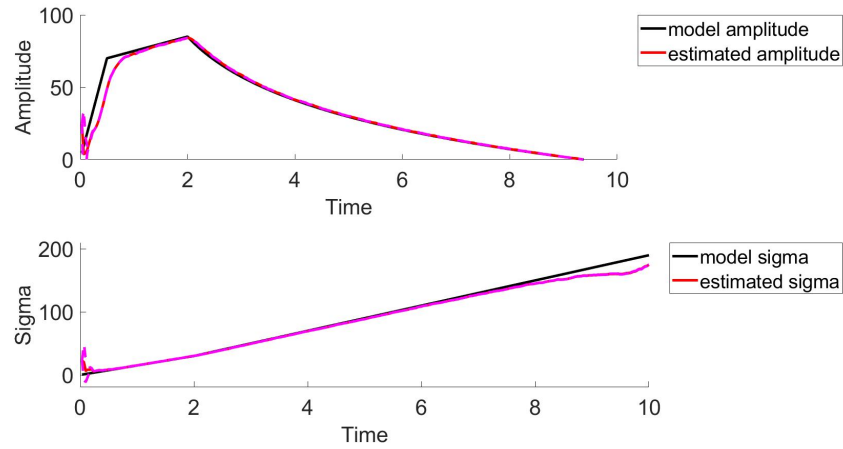


Figure 62: Estimation results for a single pulse with a standard deviation of 0.01 of random walks (A and σ). These plots display the model parameter (black), estimated parameter (red), and ± 2 standard deviation (pink).

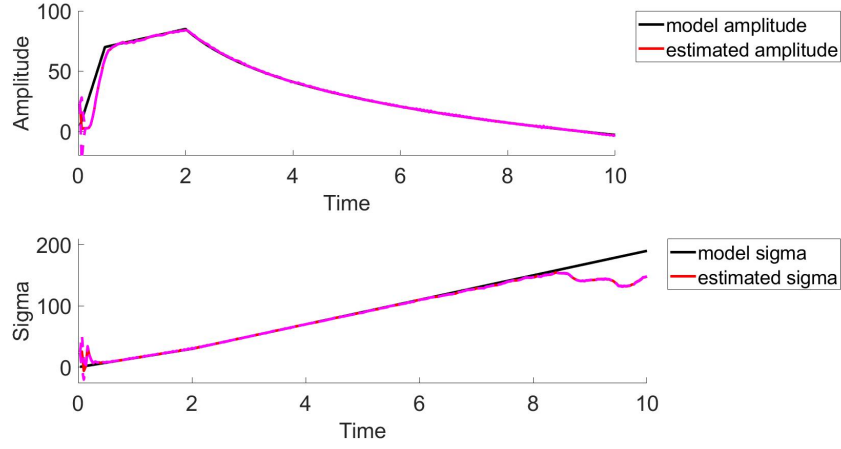


Figure 63: Estimation results for a single pulse with a standard deviation of 0.025 of random walks (A and σ). These plots display the model parameter (black), estimated parameter (red), and ± 2 standard deviation (pink).

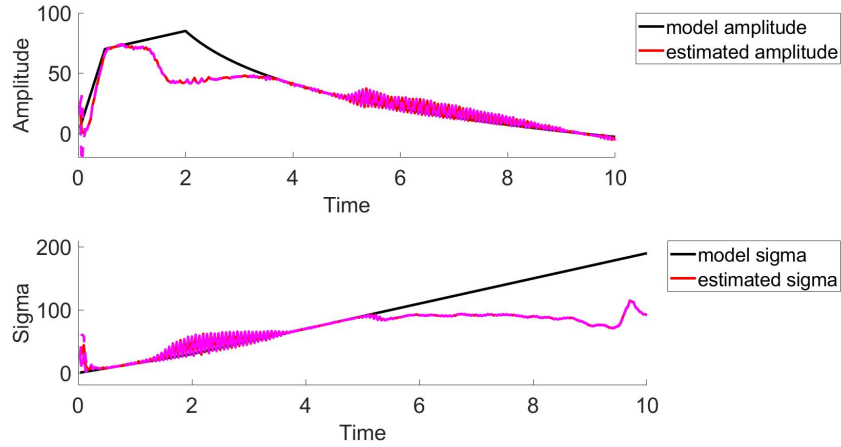


Figure 64: Estimation results for a single pulse with a standard deviation of 0.05 of random walks (A and σ). These plots display the model parameter (black), estimated parameter (red), and ± 2 standard deviation (pink).

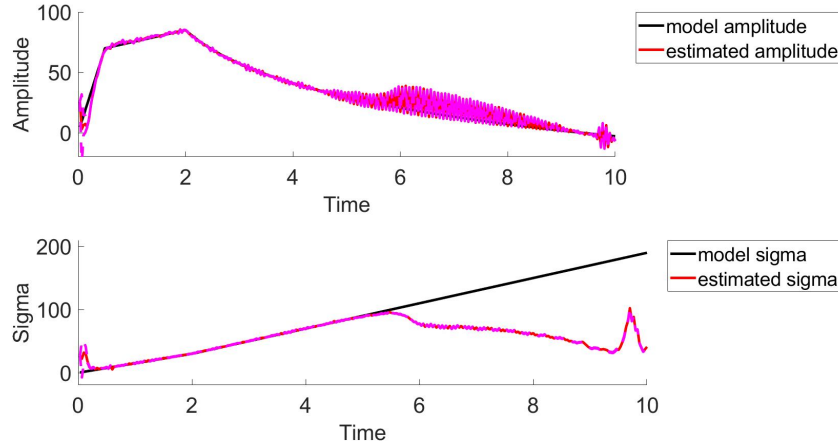


Figure 65: Estimation results for a single pulse with a standard deviation of 0.1 of random walks (A and σ). These plots display the model parameter (black), estimated parameter (red), and ± 2 standard deviation (pink).

B.4.2 Results from a Multiple Pulse Simulation

With the simulated data for multiple pulses and to be more realistic, we decided to consider the A , σ , and μ parameters as time-varying. With an additional parameter to estimate, we decided to study the effect of the standard deviation of the random walk for μ . From our observation results with a single pulse simulation, a standard deviation of 0.01 for the random walks of A and σ will result in more accurate estimations. Referring to this result, we test different standard deviation values for the random walk of μ , including 0.01, 0.001, and 0.02. Each test was conducted ten times and the results are shown in Figures 66, 67, and 68.

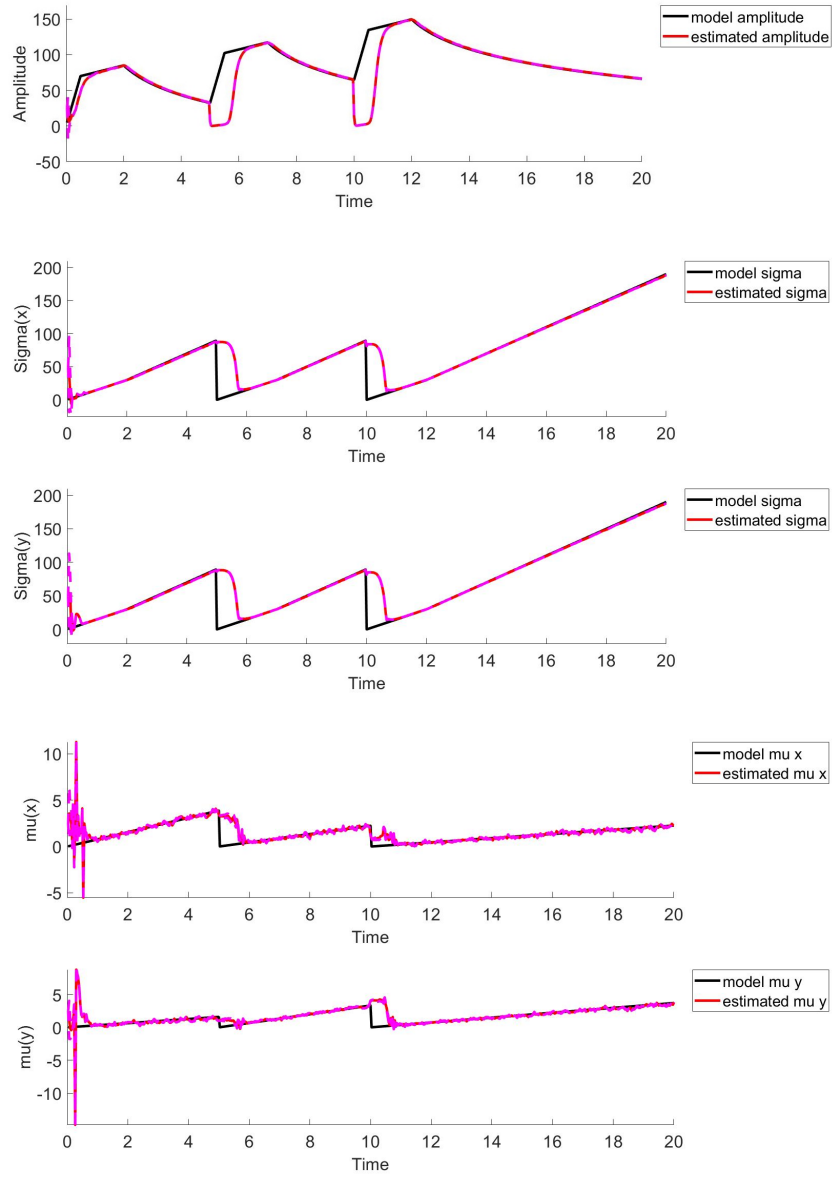


Figure 66: Estimation results for multiple pulses with a standard deviation of 0.01 for random walks (A , σ , and μ). These plots display the model parameter (black), estimated parameter (red), and +/-2 standard deviation (pink).

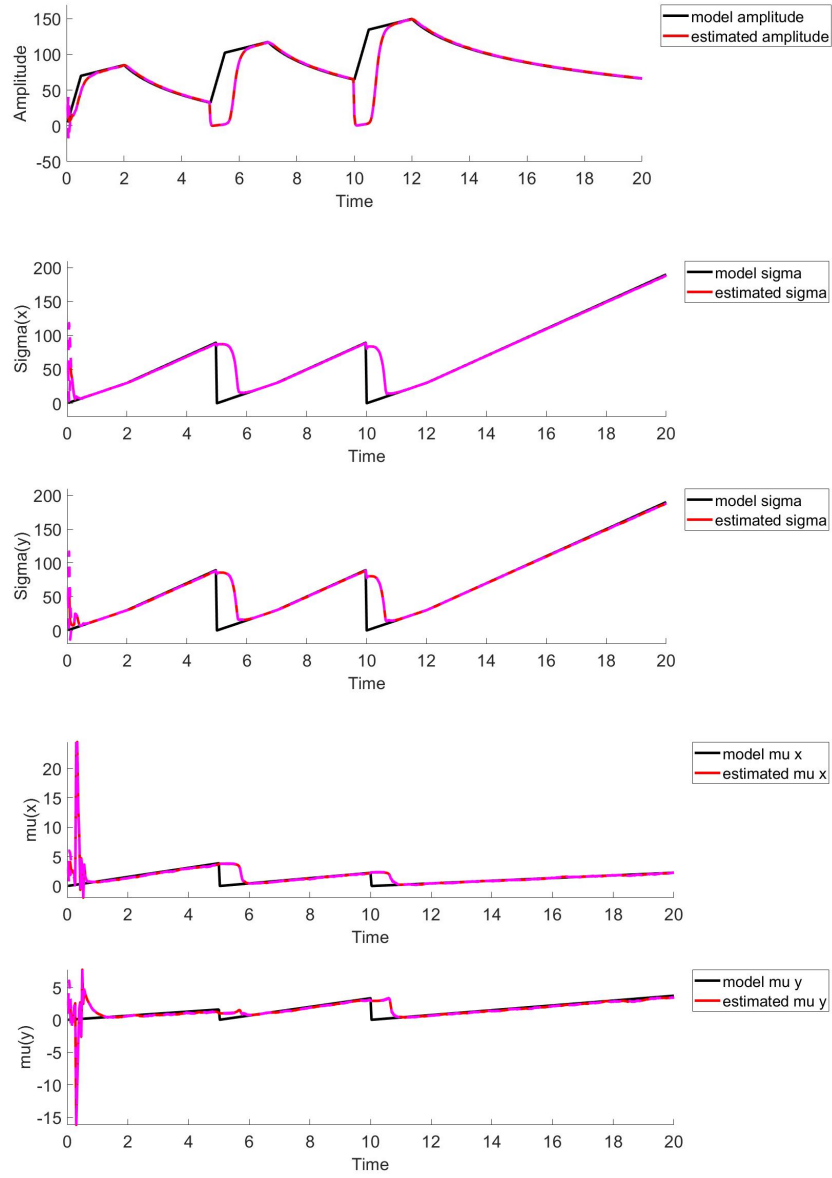


Figure 67: Estimation results for multiple pulses with a standard deviation of 0.001 for random walk (μ) and 0.01 for random walk (A and σ). These plots display the model parameter (black), estimated parameter (red), and ± 2 standard deviation (pink).

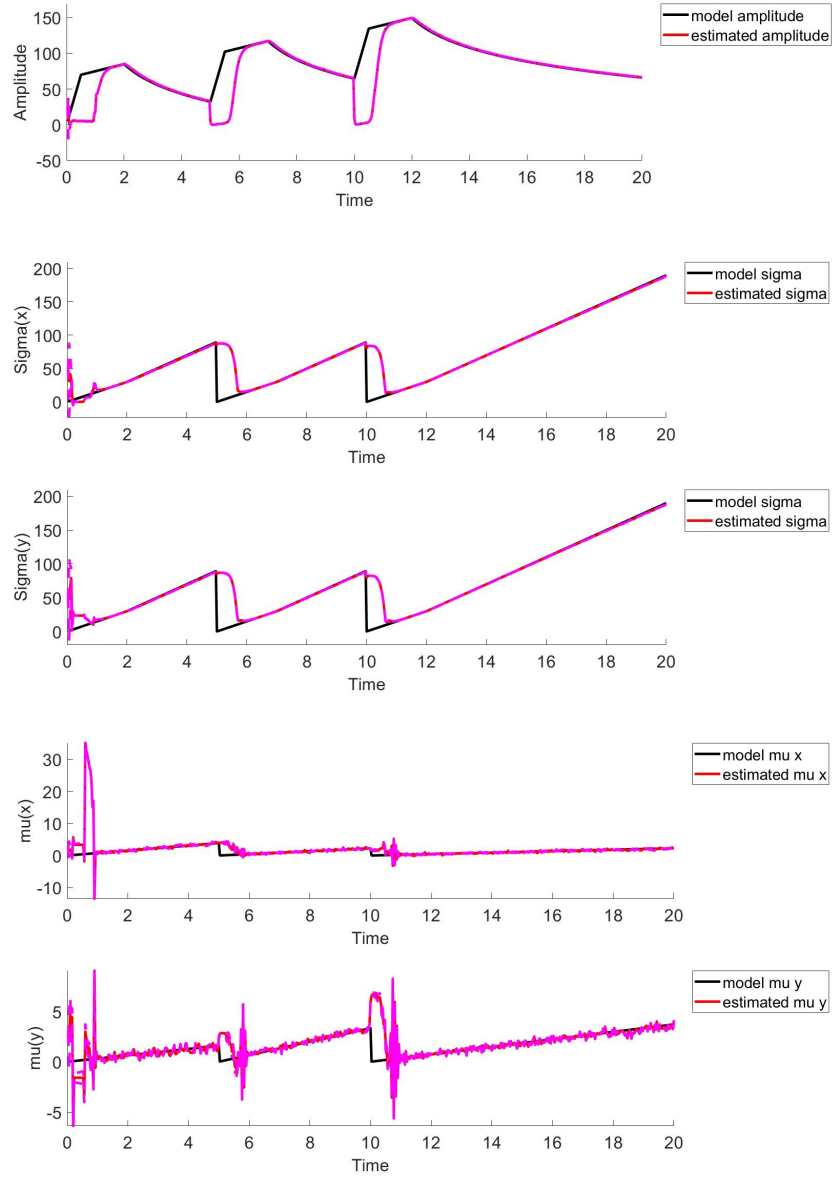


Figure 68: Estimation results for multiple pulses with a standard deviation of 0.02 for random walk (μ) and 0.01 for random walks (A and σ). These plots display the model parameter (black), estimated parameter (red), and ± 2 standard deviation (pink).

B.5 Inputs Variables of Uniform Distribution and Ensemble Size Testing

As described in Section 4.2.2, we run each of the five tests described in Table 9 five times to make observations on the effect of different input variables for the initial parameters (A , σ , and μ), as well as different ensemble sizes. Our goal is to find the input variables and ensemble size that will result in better estimations. This section records the common results of each test.

	Test 1	Test 2	Test 3	Test 4	Test 5
Lower endpoint (A and σ)	0	0	0	0	0
Upper endpoint (A and σ)	50	50	100	100	100
Lower endpoint (μ)	0	0	0	0	0
Upper endpoint (μ)	5	5	5	5	5
Ensemble size (N_{ens})	100	150	250	500	1000

Table 9: Table of the inputs of the estimation framework for the observation tests on various input values of uniform distribution (lower and upper endpoints) and ensemble sizes.

According to Figure 69, all three parameter estimations resulted in oscillations in the first 2 seconds before the estimation framework settles and becomes close to the model values. Two of the five runs oscillated for more than 2 seconds, such that the framework fails to capture the first amplitude peak. In comparison to Test 1, the μ and σ estimations for Test 2 took less than 2 seconds, displayed in Figure 70. This supports our hypothesis that a greater ensemble size will result in more accurate estimations. During the five runs, the framework is able to capture and estimate the first peak amplitude. However, some of the runs behaves similarly as Test 1, such that it oscillates rapidly in the first 2 seconds for the estimations of μ and σ .

As shown in Figure 71, the μ and σ estimations oscillates greatly during the first second, before settling to more accurate estimations. There are minor oscillations from pulse to pulse in the μ estimation. For Test 3, one of the five runs oscillated greatly for a longer time, as compared to other runs, causing it to make inaccurate estimations for the first pulse, especially on the first amplitude peak. Figure 72 shows an example of these less accurate estimations. Whereas, according to Figure 73, the results of the estimation framework (for Test 4) becomes relatively accurate relatively quickly, meaning that the estimated parameters become close to the model parameters, even though it oscillates greatly in first 1 to 2 seconds. In all of the five runs, we did not observe a low accuracy estimation, like the one shown in Figure 72, this observation is a desired improvement. Referring to Figure 74, the common estimation results for Test 5 are similar to that of Test 4. As we did not observe the presence of an essential improvement on the accuracy of the estimation to account for the additional execution time required for an ensemble size of 1000. Therefore, we concluded that the input parameters and ensemble size of Test 4 results in better performance of the EnKF framework, within a reasonable amount of time.

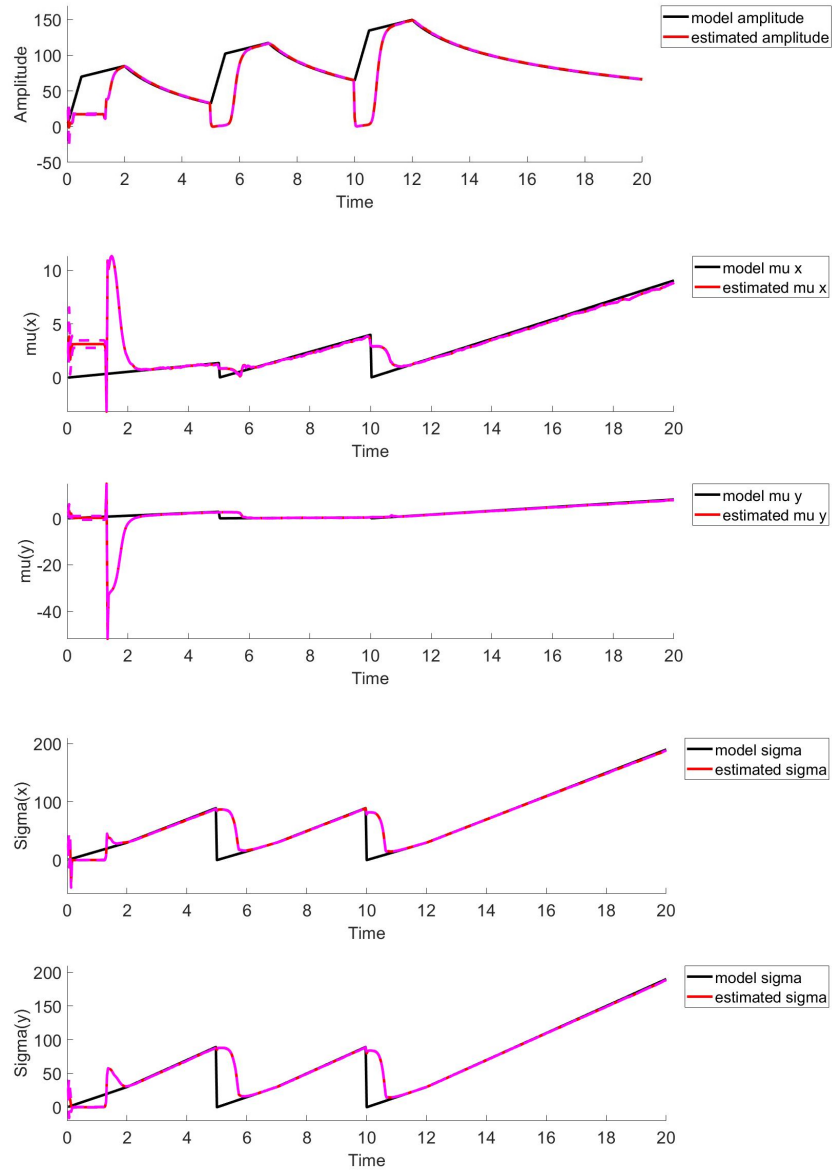


Figure 69: Common parameter estimation results for Test 1. These estimations are the common results from the five timed runs we ran for Test 1. These plots display the model parameter (black), estimated parameter (red), and ± 2 standard deviation (pink).

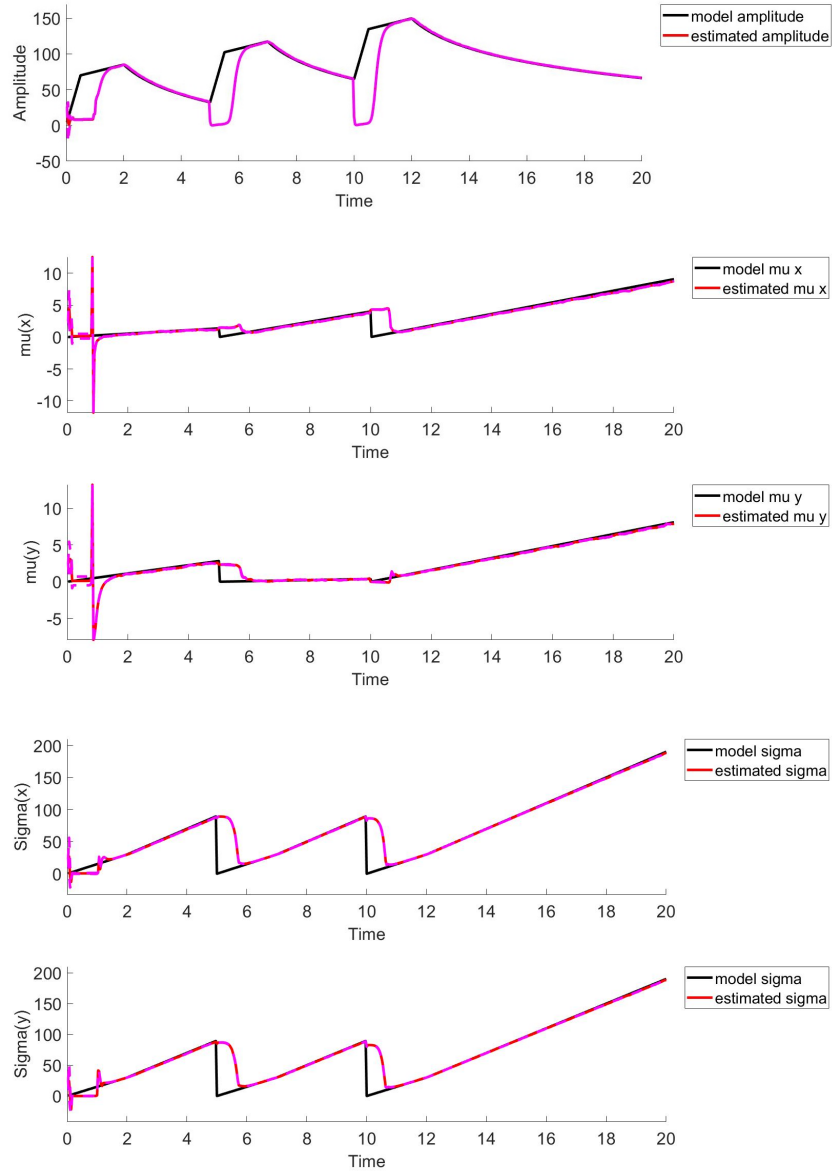


Figure 70: Common parameter estimation results for Test 2. These estimations are the common results from the five timed runs we ran for Test 2. These plots display the model parameter (black), estimated parameter (red), and ± 2 standard deviation (pink).

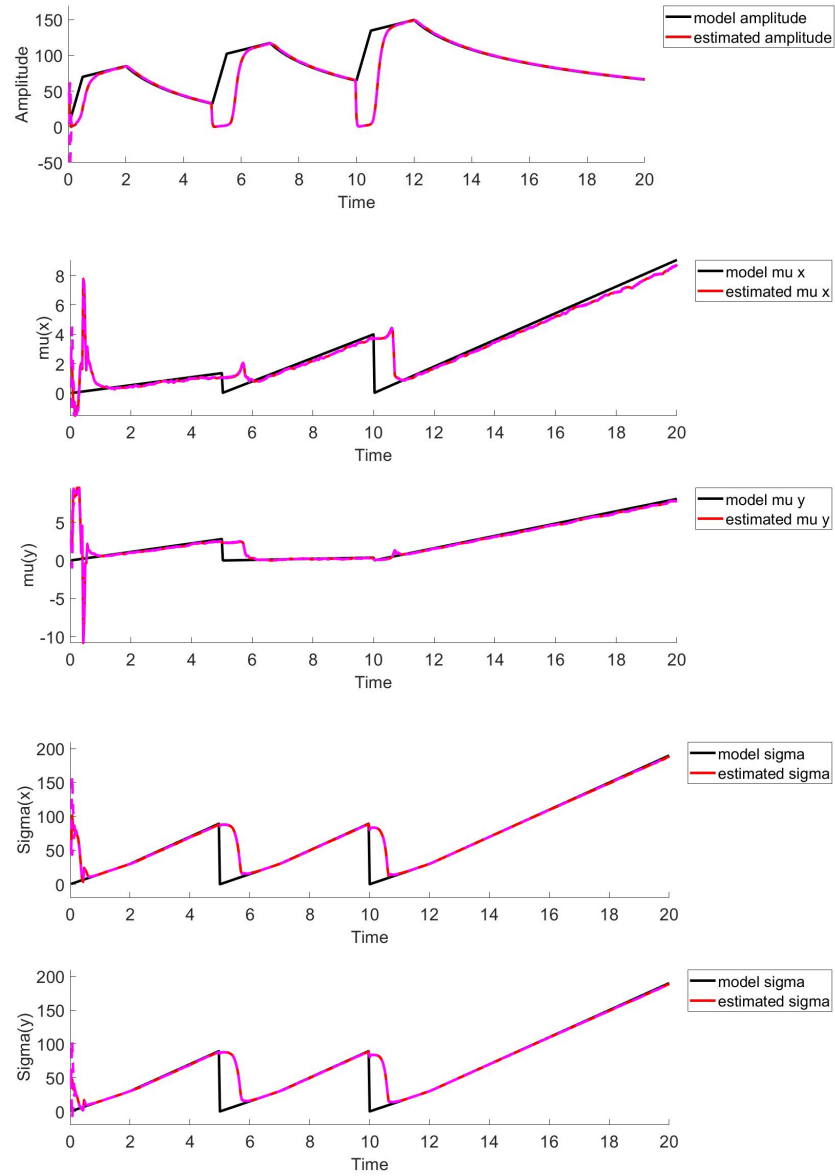


Figure 71: Common parameter estimation results for Test 3. These estimations are the common results from the five timed runs we ran for Test 3. These plots display the model parameter (black), estimated parameter (red), and ± 2 standard deviation (pink).

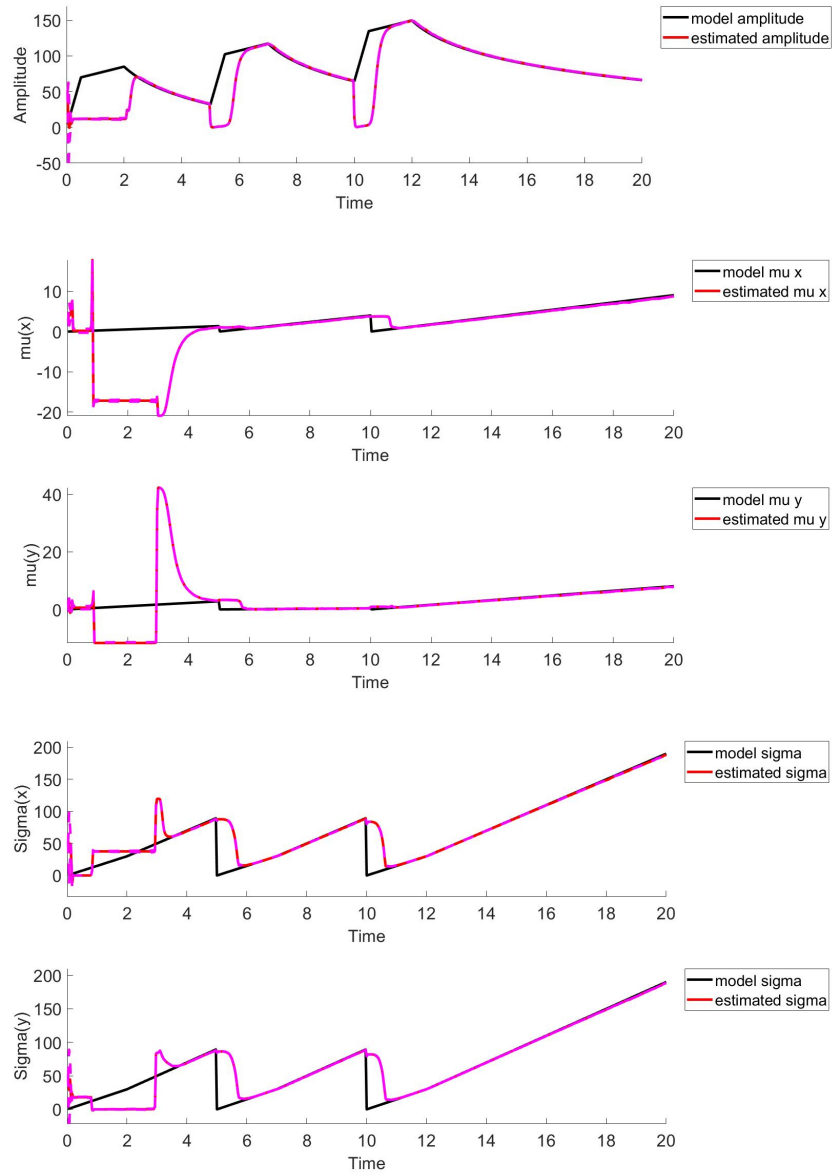


Figure 72: A less accurate parameter estimation for Test 3, which is the results of one of the five timed runs. This plot display the model parameter (black), estimated parameter (red), and ± 2 standard deviation (pink).

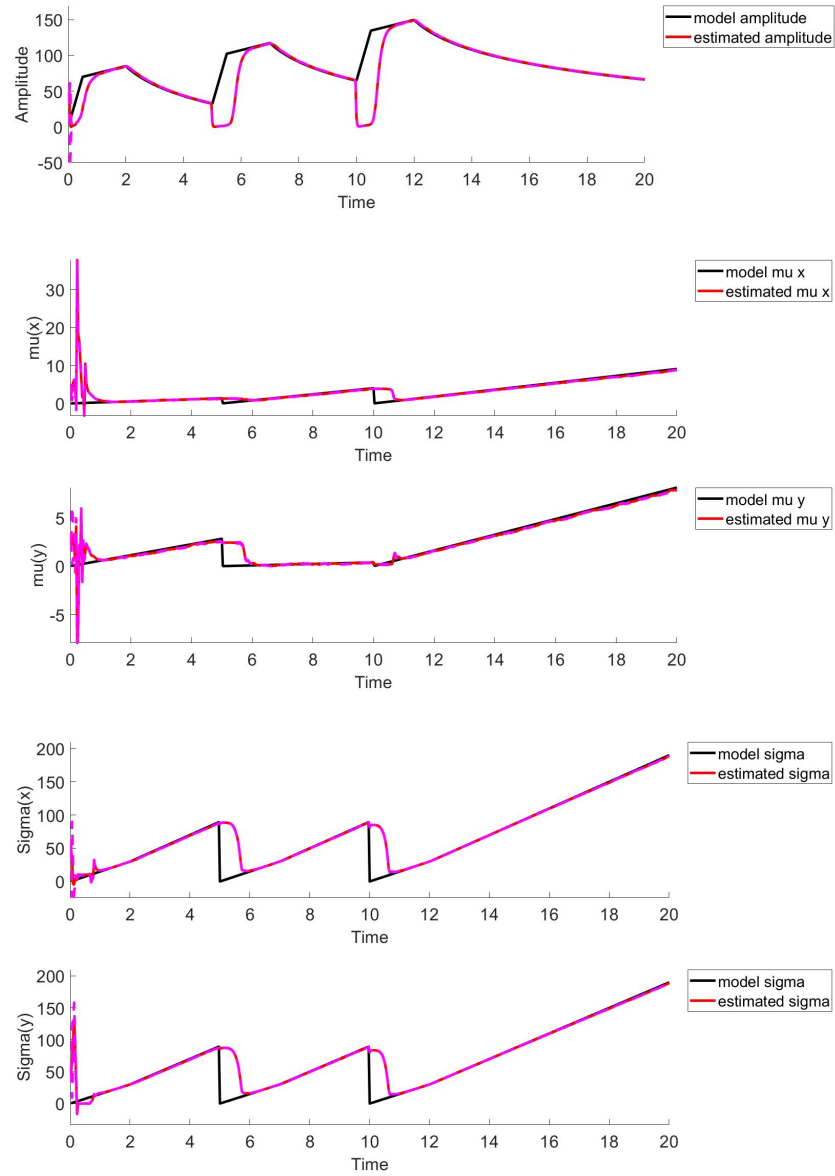


Figure 73: Common parameter estimation results for Test 4. These estimations are the common results from the five timed runs we ran for Test 4. These plots display the model parameter (black), estimated parameter (red), and +/-2 standard deviation (pink).

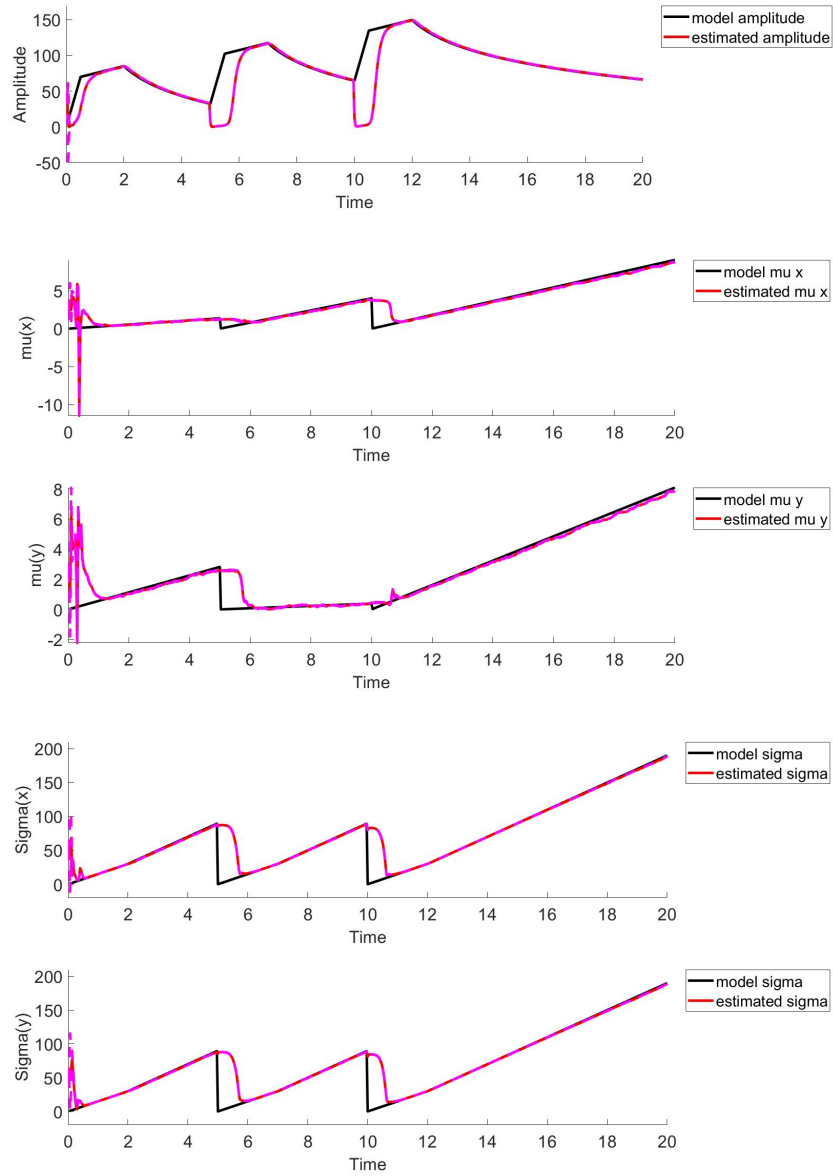


Figure 74: Common parameter estimation results for Test 5. These estimations are the common results from the five timed runs we ran for Test 5. These plots display the model parameter (black), estimated parameter (red), and +/-2 standard deviation (pink).

Appendix C Authorship Table

Section	Title	Author(s)
1	Introduction	Floris van Rossum, Yixue Wang
2	Background	-
2.1	In-Office Laryngeal Laser Procedures	Floris van Rossum
2.2	Laser-Tissue Interaction Types	Yixue Wang, Kayla Swiston
3	Methodology	-
3.1	Mathematical Modeling	Irene Wong, Yixue Wang
3.2	Computer Simulation of Sequential Temperature Data	Irene Wong
3.3	State and Parameter Estimation Framework	Yixue Wang
3.4	Machine Learning Models and Techniques for Classification of Thermal Data	Floris van Rossum
4	Numerical Results	-
4.1	Data Simulation and Parameter Estimation: Single Pulse	Irene Wong
4.2	Data Simulation and Parameter Estimation: Multiple Pulses	Irene Wong
4.3	Estimation Framework Settings with Simulated Data	Irene Wong, Yixue Wang
4.4	Classification Results	Floris van Rossum
5	Development of a Bench-top Experimental Setup	-
5.1	Data Collection Materials and Methods	Floris van Rossum, Kayla Swiston
5.2	Running Experiments to Collect Real Data	Floris van Rossum, Kayla Swiston
5.3	Preliminary Data Collection	Kayla Swiston
5.4	Experimental Data Analysis	Yixue Wang
6	Conclusion and Future Work	Everyone
-	References	Everyone
Appendix A	Mass-Spring System: An Application of EnKF	Irene Wong, Yixue Wang
Appendix B	Results of Estimation Framework Settings	Irene Wong