Rochester Institute of Technology

RIT Digital Institutional Repository

Theses

5-30-2014

An Approach for Applying Data Assimilation Techniques for Studying Cardiac Arrhythmias

Stephen T. Scorse

Follow this and additional works at: https://repository.rit.edu/theses

Recommended Citation

Scorse, Stephen T., "An Approach for Applying Data Assimilation Techniques for Studying Cardiac Arrhythmias" (2014). Thesis. Rochester Institute of Technology. Accessed from

This Thesis is brought to you for free and open access by the RIT Libraries. For more information, please contact repository@rit.edu.

An Approach for Applying Data Assimilation Techniques for Studying Cardiac Arrhythmias

by

Stephen T. Scorse

Submitted to the School of Mathematical Sciences, College of Science

in partial fulfillment of the requirements for the degree of

Masters of Science in Applied and Computational Mathematics

at the

Rochester Institute of Technology

May 30, 2014

Accepted by: Matthew J. Hoffman, School of Mathematical Sciences

Accepted by: Elizabeth M. Cherry, School of Mathematical Sciences

Accepted by: Laura M. Munoz, School of Mathematical Sciences

Committee Approval:

Matthew J. Hoffman, D.Phil. May 30, 2014

School of Mathematical Sciences

Thesis Advisor

Elizabeth M. Cherry, D.Phil. May 30, 2014

School of Mathematical Sciences

Committee Member

Laura M. Munoz, D.Phil. May 30, 2014

School of Mathematical Sciences

Committee Member

Nathan Cahill, D.Phil. May 30, 2014

School of Mathematical Sciences

Director of Graduate Programs

Abstract

Cardiac electrical waves are responsible for coordinating functionality in the heart muscle. When these waves are disrupted, it is referred to as a cardiac arrhythmia, which can be life threatening. Over the last two decades, experimental techniques to study electrical waves in the heart have been improved. Nevertheless, methods for observing electrical waves in the heart still do not provide enough data to draw conclusions about wave propagation. Observations about wave propagation can be made on the surface of the heart, but to collect data on the interior, the methods currently available either do not provide high enough spatial resolution or change the conduction properties of the region to be observed. The lack of depth information leaves us uncertain as to the precise mechanisms by which waves cause arrhythmias. Numerical models have been developed for these wave dynamics, but they are only approximations for qualitative behavior. For decades, the weather-forecasting community has faced similar problems with the approximate nature of numerical models and lack of available data, and tools known as data assimilation have been developed within that community to deal with this problem. We aim to apply these methods to cardiac wave propagation. By combining observations with numerical models through data-assimilation techniques from weather forecasting, we will attempt to constrain wave dynamics in the heart muscle interior. In this thesis, we present the first approach for coupling data-assimilation techniques with a cardiac cellular model. Using the Fenton-Karma (FK) model of cardiac cellular processes to represent wave dynamics and the Ensemble Transform Kalman Filter data-assimilation technique, synthetic observations have been integrated with the FK model. By analyzing the sensitivity of state-variable values in the FK model, we have developed initial conditions for data assimilation that are able to quantitatively approximate wave behavior in a 1-D setting. This work will provide a proof of concept and give insight into challenges in the 2- and 3-D cases.

Contents

List of Figures

List of Tables

1 Introduction

1.1 The Problem

Cardiovascular disease is the highest cause of death in the industrialized world. Approximately 10% of all deaths in the United States can be attributed to ventricular fibrillation, a fast-developing, uncoordinated contraction of the cardiac muscle that leaves the heart unable to pump blood effectively[1]. For the heart to work properly, electrical waves travel throughout the muscle. But, when the heart is not functioning properly, normal wave propagation can be disturbed, compromising the heart's ability to pump effectively. This disruption is known as a cardiac arrhythmia. During an arrhythmia, the heart can beat too fast, too slow, or with an irregular rhythm [2].

Much remains to be understood about the dynamics of arrythmias due to a lack of understanding about the behavior of electrical waves. A large part of this lack of knowledge is because the thickness of the heart muscle obstructs us from viewing the waves deep in the interior of the heart. Over the last 20 years, new high-resolution optical mapping techniques have provided more information about wave dynamics on the heart's surface, but not from within the muscle depth. Probes can be inserted into the tissue to record interior wave information, but using enough probes to properly see wave dynamics would ultimately change the behavior we hope to observe by altering the electrophysiological properties of the tissue[4]. Instead, we will modify data-assimilation techniques from numerical weather forecasting to study these wave dynamics without disrupting the behavior of wave fronts.

Forecasting a physical system generally requires both a model for the time evolution of the system and an estimate of the current state of the system. In some applications, the state of the system can be measured directly with high accuracy. In other applications, such as weather forecasting and cardiac modeling, direct measurement of the global system state is not feasible. Instead, the state must be inferred from available data. While a reasonable state estimate based on current data may be possible, in general one can obtain a better estimate by using both current and past data. Data assimilation provides such an estimate on an ongoing basis, alternating between a forecast step and a state estimation step, or analysis. The analysis step combines information from current data and from a prior short-term forecast (which is based on past data), producing a current state estimate. This estimate is used to initialize the next short-term forecast, which is subsequently used in the next analysis, and so on. The data-assimilation procedure is itself a dynamical system driven by the physical

system [20].

In this thesis we give a novel approach to constrain cardiac electrical wave dynamics in a 1-D setting. The model for our physical system that is used in this thesis is the 1998 Fenton-Karma (FK) model of cardiac cellular processes. This model can recreate the main behavior of cell dynamics with only three variables. Another model that represents the same physical system, the 1977 Beeler-Reuter model, contains 8 variables [11]. The FK model is able to replicate results of more complex models like the Beeler-Reuter model and experimental data. Computationally, the FK model is much more efficient to work with.

For data assimilation, synthetic data from the 1-D FK model will be used. There are linear and non-linear Kalman Filters. They all have two main steps in common, computing the analysis estimate (described more in section 5) and progressing the analysis forward in time for some time step. The data assimilation algorithm used for this thesis is the Ensemble Transform Kalman Filter (ETKF). ETKF allows for a hybrid approach. The FK model is non-linear. While the Extended Kalman Filter uses a linearization of the associated model and uses matrix operations to progress the analysis forward in time [13], we numerically progress the analysis forward in time using the true, non-linear FK model equations. The analysis and its associated covariance for the first iteration of the ensemble Kalman filters assume a Gaussian distribution. If we progress the analysis forward in time in a linear fashion, the associated Gaussian distribution will remain Gaussian [20]. This is not true for a non-linear time progression, but ensemble Kalman filters assume that this is the case. Even though a non-linear time progression for the FK model is used, a linear approach is used to deal with the observations. Part of the ETKF algorithm involves mapping the ensemble from the FK model space to the space of observations via a function $\mathbf{H}$, which can be linear or non-linear. We assume it to be linear since in real-world experiments, observations are separated evenly in space and time.

As far as we know, this is the first approach combining a cardiac model with a data-assimilation technique, and will help create improved estimates of wave behavior. We chose to solve the problem initially in one spatial dimension. This case is computationally much faster to solve than in higher dimensions. Successful implementations of the 1-D model can be adapted to solve the 2 and 3-D cases. We are able to gain insight to instability and sensitivity in the model in the 1-D case and can translate them to 2 and 3-D.

1.2 Outline of the Thesis

Section 2 describes the FK model that has been chosen to represent wave dynamics for this thesis. Cellular processes involved are also discussed. Here, initialization of the model as well as our method for numerically solving the model are described. Before implementing the data-assimilation algorithm, several experiments are run to gain insight into the model and advise algorithm development. The first, in section 3, involves adding error to the variables of our model. Here, we learn how sensitive each variable is to initial conditions, how fast errors grow in response to the perturbations, and how to use values that we can observe experimentally to help correct variables that cannot be seen in experiments. Based on what we learned, we then perform breeding experiments. Described in section 4, breeding is similar to the error analysis performed in the previous section, but over small time intervals. Breeding has two uses. The first is identifying areas of instability in the model, specifically time intervals where wave dynamics of the FK model cause error to accumulate faster than other times. This is done by integrating perturbed variables over a certain, shorter period of time, where at the end, differences in the perturbed and control variable values are analyzed. The second thing a breeding experiment can help with is finding a good ensemble of initial conditions for data assimilation. The data-assimilation process is discussed in section 5. Results from data assimilation with the FK model are discussed in section 6, which is followed by a brief discussion and possible future work in section 7.

2 The Fenton-Karma Model

This section provides an overview of cellular processes, wave behavior, and the model that we have chosen to represent them. Reasoning behind focusing on assimilating discordant alternans (introduced below) is given. Also discussed are parameter values, initialization and the numerical solution method of the model for this thesis.

2.1 Cardiac Background

The model we have chosen to represent the physical system is the 1998 Fenton-Karma (FK) model [8]. The FK model is a model of cardiac cellular processes that approximates the electrical properties of cardiac cells. The cells being modeled are excitable. Excitability in this context is the ability of the cell to react in response to an electrical stimulus and create an action potential. The cardiac action potential is a nonlinear response to a stimulus above a threshold in which the difference of potential between the interior and the exterior of each cardiac cell rises quickly and falls slowly back to a steady state[12]. The diastolic interval is the time when the cell is in a period of relaxation between action potentials. Restitution refers to the relationship between an action potential and the length of the diastolic interval before it. The membrane potential of the cell increasing due to an electrical stimulus is known as depolarization. Repolarization is the process by which the membrane potential of the cell returns to its resting potential. Diffusion is the ability of an excited cell to affect the excitability of the neighboring cells. This property allows for wave propagation in the heart.

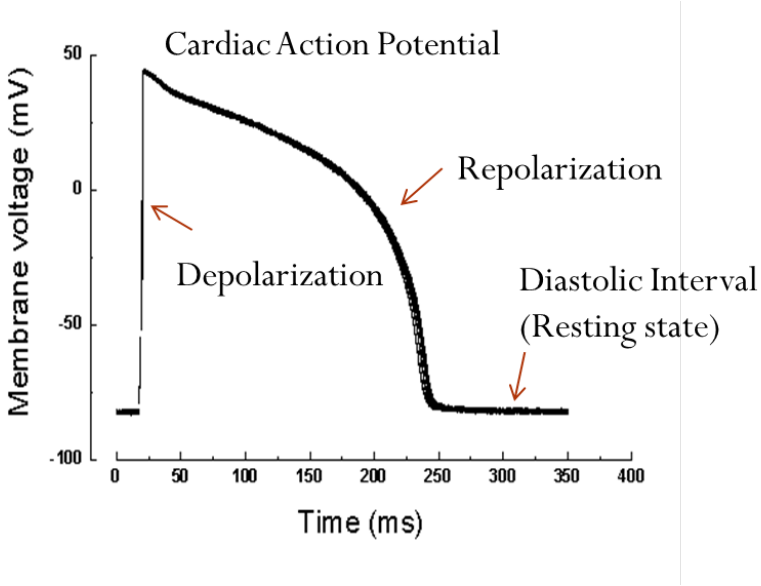


Figure 1: A cardiac action potential.

This model contains the minimum number of ionic membrane currents necessary to give restitution curves [4]. The first is the "fast inward" current, which is mainly responsible for depolarization and represents sodium ions entering the cell. The next is the "slow outward" current, which represents potassium leaving the cell and begins repolarization. The last is the "slow inward" current, that represents calcium leaving the cell. The sodium, potassium, and calcium ions follow the concentration gradient from high (outside the cell for sodium and calcium, inside the cell for potassium) to low, which gives the current directions. These ions are not tracked explicitly in this model, which is why the currents are not referred to as sodium, potassium, and calcium currents, but as fast inward, slow outward, and slow inward respectively.

The phenomenon that will be focused on during data assimilation is known as discordant alternans. When the cardiac tissue enters this state, the action potential duration (APD) in individual cells alternates between long-short and short-long in other cells; this pattern can occur with single waves. If the domain experiences APD alternating between long and short only, then this is known as concordant alternans. This APD pattern for concordant alternans is for successive waves. The domain we chose for this problem after initialization is a 1-D ring, which will be discussed later in this section. There will only be one wave on the ring. So, we must have discordant alternans as opposed to concordant alternans since the APD pattern for discordant alternans can occur with a single wave where as

the APD pattern for concordant alternans is between successive waves. Due to non-uniformity in conduction velocity restitution of electrical waves [7], the conduction velocity of the front of the wave can become faster than the back of the wave, which causes the front to catch up to the back on the domain. Their interaction causes both the front and back of the wave to oscillate, which decreases the voltage of the wave. Many studies have shown that for restitution curves with a slope greater than 1, unstable behavior, including alternans, can occur [4, 8, 7, 11].

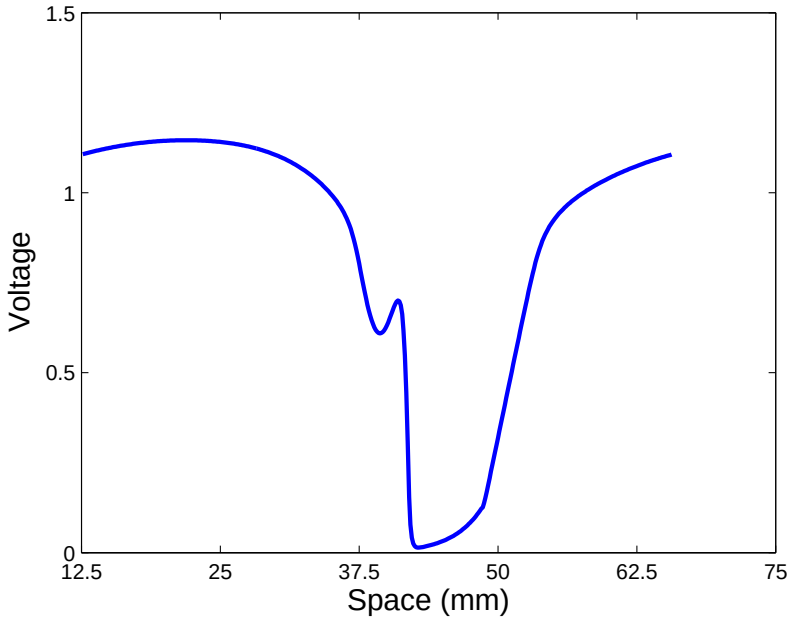


Figure 2: Interaction between the wave front and back due to discordant alternans.

Figure 2 shows just after an interaction of the front and back of the wave. The front is located around 40 mm, while the back of the wave is roughly between 50 and 55 mm. The voltage of the front has been cut approximately in half, around 40 mm. This voltage change here will transiently affect the voltage of the rest of the wave shortly after.

2.2 The Model

Consisting of only three variables, this model is very simplistic, yet is able to capture the main dynamics of cell excitation and is able to reproduce results similar to more complex ionic models as well as results from experimental data [5, 6]. The Beeler-Reuter model that represents the same physical system, for

example has 8 variables [11]. Other models that represent human ventricular cells like the Priebe-Beuckelmann, Tusscher-Noble-Noble-Panfilov, and Iyer-Mazhari-Winslow models, have 22, 17, and 67 variables respectively [9]. Other models also exist. This model is a good fit for maintaining the balance between computational efficiency and representing properties of cardiac tissue; in particular, it reproduces discordant alternans sufficiently. The equations of the model are as follows:

$$\partial_t u = (D\partial_{xx} u) - J_{fi}(u;v) - J_{so}(u) - J_{si}(u;w), \tag{2.1}$$

$$\partial_t v = \frac{\theta\left(u_c - u\right)\left(1 - v\right)}{\tau_v^-} - \frac{\left(1 - \theta\left(u_c - u\right)\right)v}{\tau_v^+}, \tag{2.2}$$

$$\partial_t w = \frac{\theta\left(u_c - u\right)\left(1 - w\right)}{\tau_w^-} - \frac{\left(1 - \theta\left(u_c - u\right)\right)w}{\tau_w^+}. \tag{2.3}$$

The dimensions for the three derivatives are are inverse time. u represents the membrane potential, the difference in electrical potential inside and outside of a cell. u is bounded below by 0 and in these simulations, a maximum value of approximately 1.5 is seen. v and w are gating variables, which measure the cell membrane's permeability to let specfic ions pass through the cell membrane. A value of 0 means no ions can pass, while a value of 1 means the ion channel is at its maximum permeability. Homogeneous Neumann boundary conditions are imposed on u for this model. All τ parameters are time constants. See Table 1 below for parameter values and definitions. u_c is a membrane potential threshold that changes the active currents in or out of the cell. θ represents the Heaviside step function. Mathematically, u_c is used in the Heaviside function to make each equation piecewise. D is the 1-D diffusion coefficient, which when combined with u_{xx} allows for wave propagation. Each J is a membrane current:

$$J_{fi}(u;v) = -\frac{v}{\tau_d}\left(1 - \theta\left(u_c - u\right)\right)\left(1 - u\right)\left(u - u_c\right). \tag{2.4}$$

$$J_{so}(u) = -\frac{u}{\tau_o}\theta\left(u_c - u\right) + \frac{1}{\tau_r}\left(1 - \theta\left(u_c - u\right)\right) \tag{2.5}$$

$$J_{si}(u;w) = -\frac{w}{2\tau_{si}}\left(1 + tanh\left[k\left(u - u_c^{si}\right)\right]\right). \tag{2.6}$$

The dimensions for these three equations are also inverse time. J_{fi} is the "fast inward" current, J_{so} is the "slow outward" current, and J_{si} the "slow inward" current.

2.3 Parameter Values

The following parameter values were used in solving the model. They are reproducing the Beeler-Reuter model [10] on a 1-D cable because they are conducive to creating discordant alternans [8].

Parameter	Meaning	Value	Units
dx	grid step size	0.25	mm
$nstart$	number of grid points	500	1/mm
D	1-D diffusion coefficient	0.001	$\frac{\text{cm}^2}{\text{ms}}$
τ_r	governs late repolarization	33.33	ms
τ_o	governs early repolarization	12.5	ms
τ_{si}	controls action potential duration	29.25	ms
τ_v^+	controls recovery time from an action potential	3.33	ms
τ_w^+	controls the maximum action potential duration	870	ms
τ_w^-	controls amplitude of action potential	41	ms
u_c	threshold potential for activation of J_{fi}	0.13	1
u_v	threshold potential for $\tau_v^-(u)$	0.04	1
u_c^{si}	threshold potential for activation of J_{si}	0.85	1
k	controls steepness and width of action potential	10	1
c_m	membrane capacitance	1	$\frac{\text{microFarads}}{\text{cm}^2}$
g_{fi}	maximum conductance of the sodium (fast inward) channel	4	$\frac{\text{milliSiemens}}{\text{cm}^2}$
τ_d	controls excitability	c_m/g_{fi}	ms
τ_{v1}^-	controls steepness of wave front	1250	ms
τ_{v2}^-	sets the minimum diastolic interval	19.6	ms
$\tau_v^-(u)$	governs reactivation of the fast inward current	$\theta(u-u_v)\tau_{v1}^- + \theta(u_v-u)\tau_{v2}^-$	ms

Table 1: FK model parameters used in this study.

2.4 Initialization of the Model

In order to reproduce discordant alternans, a complex procedure is required to initialize the model on the ring domain. Using the parameter values chosen above, we initialize the system as a 1-D cable with a uniform grid of 500 points, and with $u = 0.2$ on the first half of the grid, $u = 0$ on the second half of the grid, and $v = w = 1$ for all grid points. This means that at time $t = 0$, there is no membrane potential difference on the second half of the grid and ion channels are allowing as many ions that it physically can to pass through the cell membrane. This step function initialization of u allows a wave

to begin forming on the domain. The 500-point grid was found to be the smallest to allow the wave created from the initial conditions to become stable. Since the boundary conditions on a cable are homogeneous Neumann, a propagating wave would terminate on the boundary, and we would need to continually provide an external stimulus to the system, which could change the wave behavior that we are trying to observe. After the initial stimulus, two waves propagate in opposite directions. When one reaches the boundary and terminates, we switch the domain to a 1-D ring.

After 25 ms, we switch to a ring. If the domain is a ring of length L, then at time t, $u(t,0) = u(t,L)$. This means that a wave can essentially travel through the boundary, eliminating the need for an external stimulus to sustain wave propagation. After the wave adjusts to the domain size, we shrink the ring size over discrete time steps. The domain shrinking changes the period of the wave on the ring, which helps the system reach a state of discordant alternans. The final number of grid points is 425 for the parameter set used. While shrinking, we must be careful to not delete parts of the ring that contain the front of the wave, as this would most likely cause the remaining part of the wave to annihilate. We check for this every 375 ms by making sure $u < .1$ on the part we want to remove. If the wave is not located at the end of the domain, 15 grid points are removed, otherwise we wait another 375 ms to check again. After some testing, it was decided that 375 ms gave the wave enough time to become stable on the new domain size and 425 was set as the final domain size because anything smaller led to wave annihilation, leaving $u = 0$ at all grid points.

Once discordant alternans is seen at the final grid size and stabilizes, we progress the model for an additional 3000 ms. This time series serves as a truth for data assimilation and a control for breeding. We focus on implementing the data-assimilation process during discordant alternans because it is less deterministic than other states of the model and it is therefore a more challenging case to approximate solutions during this dynamical state.

2.5 Numerically Solving the Model

Second order finite difference quotients were used to integrate the system numerically using the forward Euler method. Initially, the model was solved on a 1-D cable, and then switched to a ring geometry after initial wave startup for the reasons described early in this section. Solving the model was implemented in programs written in Fortran 90 while visualization of the solutions was done with Matlab 2012b.

A good compromise between computational accuracy and data storage needed to be found. By increas-

ing the time and grid step sizes, array sizes decrease, reducing the required storage space. Because of this, time steps were varied from 0.1, 0.2, 0.25, 0.3, and 0.4 ms. 0.1 ms did not cause much change in wave behavior when visually compared with 0.2 and 0.25 ms, while 0.3 and 0.4 ms caused the system to become numerically unstable. The final time step settled on was 0.25 ms since it visually compared well with 0.1 and did not cause instabilities.

3 Analyzing Error and Instability Growth of the FK Model

This section covers our method for evaluating the instabilities and error in the FK model. The importance of results for setting up breeding experiments is discussed. Results from this method are also given.

3.1 Method and Overview

Before running breeding experiments or data assimilation, information on the model's sensitivity to error in conditions, namely variable values, is needed to help create guidelines for the breeding experiment's time intervals and error perturbations. To achieve this, the model is run with the initial conditions and parameter values stated above. This is called the control state. When the model enters a state of discordant alternans, combinations of control state values of u, v and w are taken and random Gaussian error with a mean of 0 and standard deviation of 0.05 is added to all points in the domain. We begin perturbations here, and at this stage of our approach, only at this time, to isolate errors caused by grid shrinking and the alternans. Each time the grid size is reduced, the wave front can catch up to the back faster than it could on the previous grid size due to the change in the period of the wave. Until the wave adapts to the new, smaller domain, alternans can become more frequent. These perturbations are then added to the variable values at the time desired as long as they stay within bounds. For example, it does not make physical sense for any of the variables to be negative, so if one of the perturbations caused any of them to become negative, a new perturbation is selected. Other restrictions include keeping $v, w \leq 1$ and $u \leq 1.5$. The model is then run to the final time, which at this stage was 2500 ms, and differences in the control and perturbed states are noted for all variables, over all grid points, over time. These differences are examined using the root mean square (RMS) difference between the control and perturbed states.

A small change in these conditions can completely change the wave's length, shape and voltage over time. These traits of the wave can look very different at the same time in the model under different perturbations. Consider a perturbation in the w variable. A small perturbation added to every grid point from the mentioned Gaussian curve causes the wave back to look very different after 700 ms (Figure 3). The two waves shown are spatially shifted. The control state has just experienced a state of discordant alternans while the perturbed state appears to be close to entering a state of discordant

alternans since the wave front (located near 50 mm) is close to the wave back (centered around 80 mm).

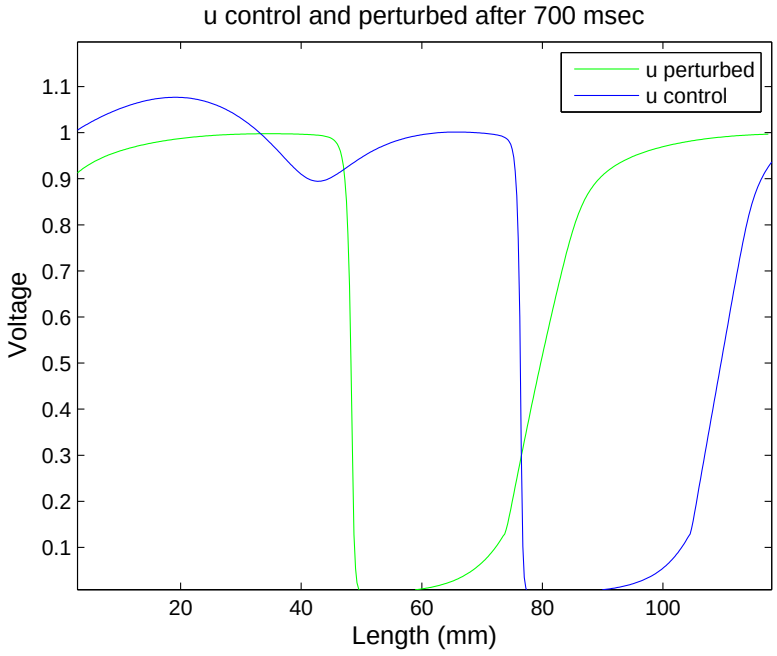


Figure 3: Examples of the control and perturbed states of u 700 ms after a perturbation in w in part of the domain.

3.2 Instabilities of the FK Model

Perturbations in u resulted in error growth similar to what is shown in Figure 4. After some initial spinup, or transient behavior, from the perturbation, error of u and v increase linearly and then plateaus. They could possibly be increasing logarithmically. The error of w remains small and nearly constant. In some cases, the RMS error did not oscillate as much. The magnitude of error for u and v is approximately 3 times as high as the error for w. Perturbations in v resulted in plots similar to those from u perturbations.

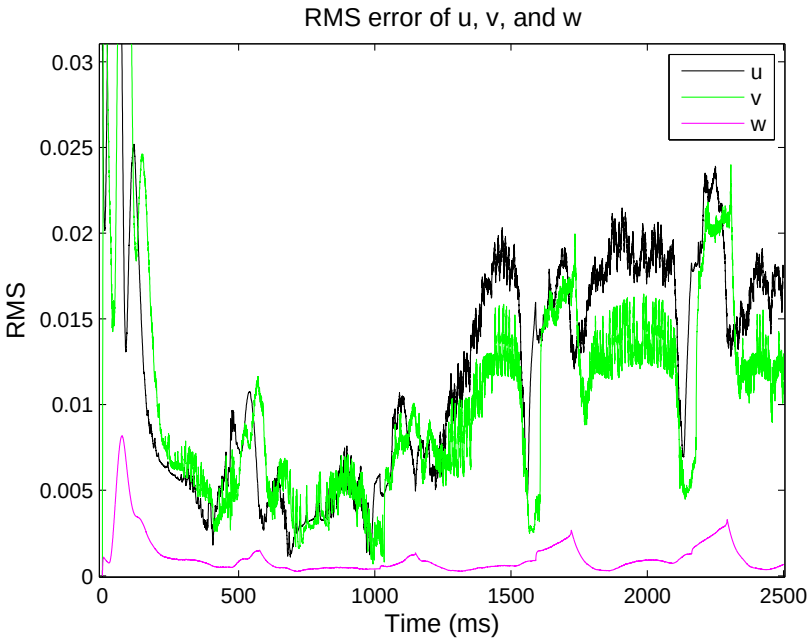


Figure 4: Differences between the control and perturbed states of all variables over time, after a perturbation in u.

When w is perturbed, the errors of u and v become oscillatory, with longer oscillation period and larger amplitude than when u is perturbed. Errors decrease linearly for u and v, while w oscillates around a constant value. Again the oscillations for w in this case are larger than when u was perturbed. The errors trend in Figures 4 and 5 to roughly the same values for all three variables by 2500 ms.

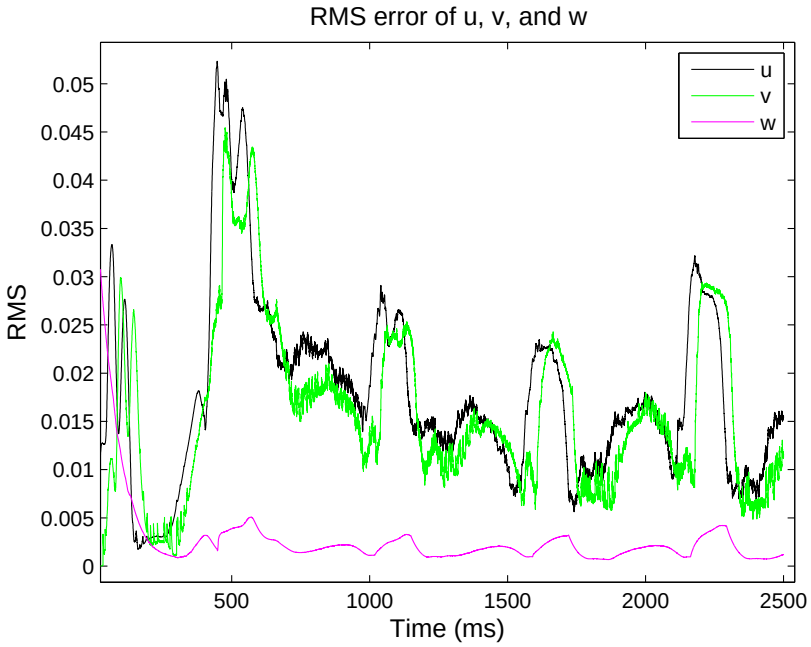


Figure 5: Differences between the control and perturbed states of all variables over time, after a perturbation in w.

When all three variables were perturbed, the RMS plots (see Figure 6) looked very similar to Figure 5, where only w was perturbed. The order of magnitude and oscillations of errors in u and v all look very alike in both cases. The plot suggests changes in w could be dominating the changes in u and v.

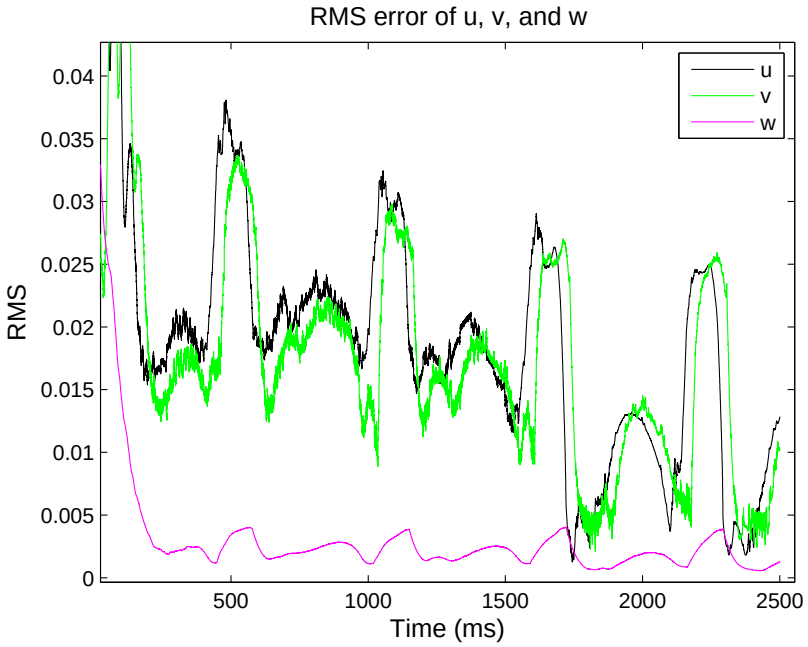


Figure 6: Differences between the control and perturbed states of all variables over time, after a perturbation in all variables.

Similar results in all cases were seen over multiple simulations. Error in initial conditions do not grow quickly, as seen in Figures 4, 5, and 6, and tend to decay or trend toward constant values. Errors in different variables appear correlated, indicating that observations from one variable can be used to correct errors in another (unobserved variable). For example, we cannot directly observe values of the v and w variables experimentally. Our results in this section suggest that the errors in u and v are correlated. Errors in v seems to mimic errors in u with a slight delay and less magnitude. This can be used to our advantage. We now know that it may be possible to use the u variable to make corrections in v.

4 Breeding Experiments on the FK Model

Here we discuss what a breeding experiment entails and how its results can be used to identify temporal and spatial locations of instabilities of the parameter value set used for the FK model.

4.1 The Breeding Process

The breeding method is used to identify regions of the domain and times of a dynamical system where error grows rapidly in the model variables, which in turn could help provide reasonable initial conditions during the data-assimilation phase. When the FK model enters a state of discordant alternans and after the domain has reached its final size, we begin the breeding experiment.

A breeding experiment can be outlined as follows:

1. Add a perturbation to the control state at some time, call it t_0.
2. Integrate both the control state and perturbed state until some time $t_1 > t_0$. The length of time between t_0 and t_1 is called the breeding interval.
3. At time t_1, subtract the control state from the perturbed state for all grid points and rescale the difference, so that it has the same norm as the initial perturbation. The norm used here is the RMS.
4. This rescaled difference is then used as the new perturbation to the control state.
5. Repeat steps 1-3 [15].

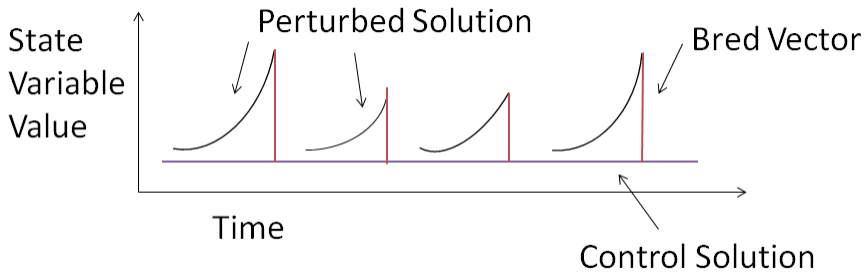


Figure 7: The breeding process.

The iterated difference between the perturbed and the control solution is called the bred vector and

is illustrated in Figure 7. The system we have created now has random error added into it via the perturbations. Dynamical systems have stable and unstable subspaces associated with them. Random error added into the FK model will be projected onto these stable and unstable subspaces. Errors projected onto the stable subspace will decay, while errors projected onto the unstable subspace will grow. Growing errors can be attributed to instabilities in waves. When creating ensemble members in the data-assimilation process, these growing errors are crucial [14]. It has been found that the fastest growing combinations of possible errors in perturbed states (these correspond to the largest resulting bred vectors) give the best results as initial ensemble perturbations in data assimilation, at least in the short range [18].

Rescaling is done in a deterministic fashion to help eliminate stochastic components of the perturbations. The bred perturbations are thus based solely on the dynamics of the FK system [14].

4.2 Breeding with the FK Model

Breeding could begin at any time, but we begin breeding at the final domain size and during discordant alternans because we want to further isolate instabilities from the alternans and the grid shrinking. For this phase we extended the final run time to 15000 ms to be sure that the discordant alternans did not cause the wave to annihilate completely. For data storage purposes, we only performed breeding for 3000 ms, between 6000 and 9000 ms as opposed to breeding from 6000 to 15000 ms. We begin at 6000 ms because this is shortly after we finish shrinking the grid, so it is here that effects from the alternans can be identified. Starting at anytime after 6000 ms and breeding for 3000 ms would yield the similar results since the alternans continue on the final grid size until the final time of 15000 ms.

Figure 8 shows growth rates of u, v and w during the breeding process. The growth rate we used was the RMS of the bred vector each variable at the beginning of the breeding interval divided by the RMS of the bred vector at the end of the interval. Here, a breeding interval of 10 ms was used. This figure also confirms that u and v are correlated. The growth rates of v look very similar to u and again are slightly lagging behind u. w appears to be robust as it does not change much from perturbations. This is seen in its low growth rates. The breeding results presented in this section are all consistent with the results of the previous section on error analysis and instability growth. The similar shapes of the growth rates in u and v over each breeding interval tell us that this system is not necessarily chaotic, it could, however, be weakly chaotic.

Breeding intervals of 5, 10, 25, 50 and 100 ms were also used. Results were very similar to those presented in Figure 8. The only notable differences were in time resolution.

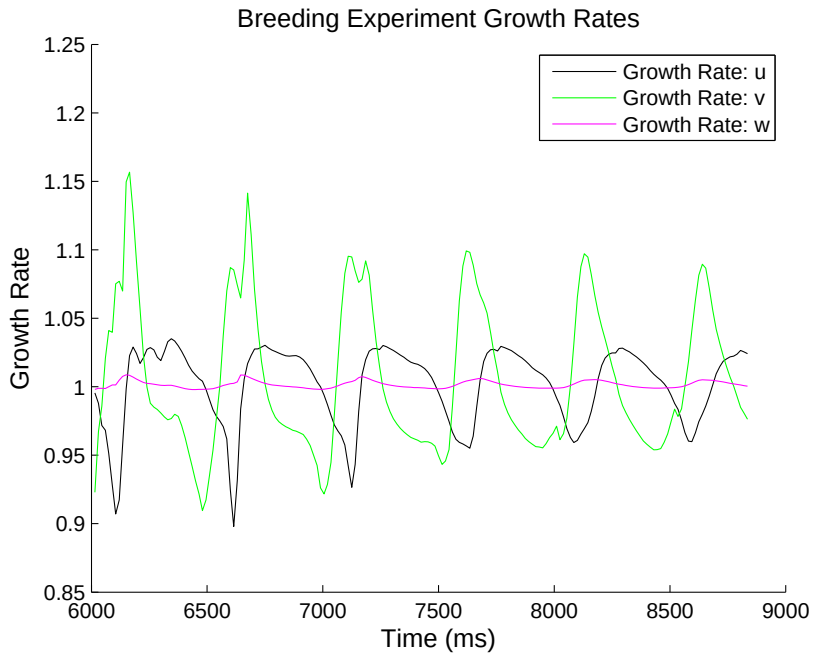


Figure 8: Growth rates from the breeding process with a breeding interval of 10 ms.

5 An Introduction to Data Assimilation

This section provides a data assimilation example from weather forecasting and connects it to our cardiac wave problem. A consequence of data assimilation, parameter estimation, is also discussed. Table 2 below gives brief descriptions of the variables and parameters as well as possible values for them used in the simulations for this thesis. More in depth descriptions of each are given throughout this section and section 6. The ETKF algorithm used to obtain our results is given last.

5.1 Background

Data assimilation is widely used to study and forecast geophysical systems. As a typical example, consider the formation of Hurricane Sandy. Due to model errors and chaotic dynamics, different numerical models can produce different paths for storms (Figure 9). While the solid black path is the actual path of Sandy, the colored paths are all different forecasts that were created 7.5 days before Sandy hit New Jersey. When different forecasts are plotted together like this, it is called a spaghetti plot. Model runs 9 to 10 days before Sandy struck predicted it would hit somewhere in the Northeast. As the model took in more data from weather balloons, weather stations, and satellites, predictions improved. Even 5 days out, however, some models predicted that Sandy would turn out into the Atlantic and never hit land. The best forecasts came from the European Centre for Medium-Range Weather Forecasts (ECMWF). Though ECMWF is usually superior and made the best predictions in this case, that is not always true. When tropical storm Debby was forming, ECMWF predicted that Debby would head to Texas. The U. S. Global Forecast System, however, correctly predicted that Debby would go to Florida [19]. This example tells us that algorithms for forecasting are sensitive to initial conditions and available data, but are not always accurate.

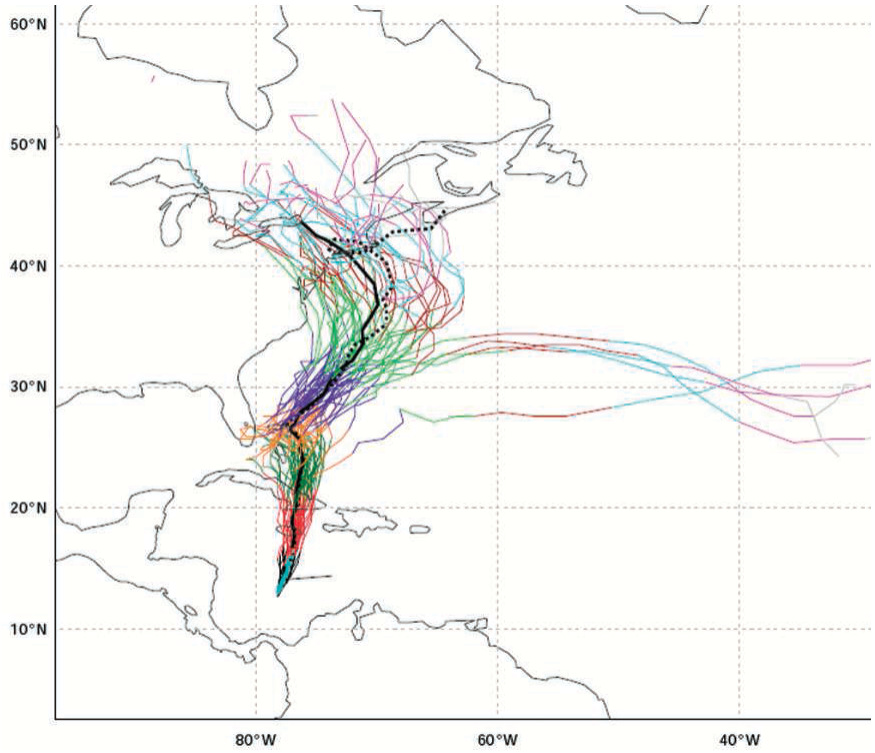


Figure 9: Model predictions for the path of Hurricane Sandy[19].

The same forecasting idea of constraining storm paths is being applied in this thesis to constrain cardiac electrical wave dynamics. For portions of the wave deep enough into the tissue, to the point where it can no longer be mapped using optical mapping techniques, we are testing to see if data assimilation can help constrain the dynamics of the waves, giving a better idea of what they look like without disturbing them.

We will be using an Ensemble Transform Kalman filter (ETKF) approach to this data assimilation problem. This method uses Gaussian distributions for the ensemble members and tracks their means and covariances over time while perturbing system states. The ensemble can be small or large, but typical sizes range from 10 to 100 ensemble members. The ensemble should be large enough to approximately span the space of growing errors at a given time, because the analysis essentially determines which linear combination of the ensemble members forms the best estimate of the current state, given the current observations. Our method used here is a modified version of the Local Ensemble Transform Kalman Filter (LETKF). It is most similar to ETKF and the Local Ensemble Kalman Filter. Results produced by LETKF are nearly equivalent to the Local Ensemble Kalman Filter, but it achieves these

results in a manner more similar to and more efficient than ETKF[20]. Here, we will use the LETFK algorithm, but skip the localization step. Localization helps determine behavior at smaller sections of the domain, but because the wave behavior that we see is mostly smooth, not very chaotic, and overall global, observations at grid points relatively far away from other points on the domain can aid in determining the wave behavior in other spatial locations, beyond a local scope. Computationally, localization promotes parallelization by allowing the algorithm to analyze different parts of the domain simultaneously. Because our domain is relatively small, localization will not help much as far as computation is concerned. The resulting process will be an ETKF approach. The localization step will most likely be necessary when the problem is scaled to 2- and 3-D.

One application of data assimilation is parameter estimation. Typically in models, there are parameter values that can reproduce certain behaviors. For example, the parameter set given in table 1 reproduces discordant alternans. But, how do we know those are the best parameters to be using? Since parameters cannot be observed experimentally, there is not a good answer to this question. By including some parameter values into our data-assimilation dynamical system as initial conditions, we hope our system can help estimate what good parameter choices will be.

5.2 The LETKF Algorithm

5.2.1 Setup

Parameter/Variable	Meaning
m	dimension of the model space
ℓ	dimension of the observation space
k	ensemble size
ρ	multiplicative covariance inflation factor
$\mathbf{X}^b_{m\times k}$	background ensemble in the model space
$\bar{\mathbf{x}}^b_{m\times 1}$	background estimate in the model space
$\mathbf{X}^a_{m\times k}$	analysis ensemble
$\bar{\mathbf{x}}^a_{m\times 1}$	analysis estimate
$\mathbf{y}^o_{\ell\times 1}$	observed values
$\mathbf{H}_{\ell\times m}$	map from the model space to the observation space
$\mathbf{R}_{\ell\times \ell}$	covariance matrix for observation space
$\mathbf{Y}^b_{m\times k}$	background ensemble in the observation space
$\bar{\mathbf{y}}^b_{\ell\times 1}$	background estimate in the observation space
$\tilde{\mathbf{P}}^a_{k\times k}$	analysis covariance matrix
$\mathbf{W}^a_{k\times k}$	matrix of weight column vectors

Table 2: Variable and parameter descriptions for the ETKF algorithm.

The algorithm uses an initial background ensemble with k ensemble members. Let $\mathbf{x}^{b(i)}(i = 1, 2, \ldots k)$ represent each of these members, each of which for our purposes is a column vector with u, v, and w appended to each other. Since the domain size of the FK model after we finish shrinking the grid is 425 and we are using each of the three variables, $m = 3 * 425$, and each $\mathbf{x}^{b(i)}$ is $m \times 1$. The initial ensemble members are obtained from our breeding experiments for the first iteration. These members need to have some sort of error in them, otherwise they would all be copies of each other, eliminating the benefit of an ensemble. If we created an initial ensemble from random perturbations, the error built into our ensemble would have little dependence on the FK model. The breeding experiment, however, allows an initial random perturbation to progress through the model for some time step and thus the initial perturbations becomes model dependent. Toth and Kalnay showed that two-member bred ensembles matched experimental observations better than similarly sized ensembles with random initial perturbations [14].

Figure 10 below is a spaghetti plot of u ensemble members to begin the data-assimilation process. These are all different perturbations of the control at 6100 ms after the start time of the model and 100 ms after the perturbations are added. One way to measure how close the members are to one another is through spread, the RMS distance each member is from each other. The members in Figure 10 have similar dynamics at this point in time. Since numerical models capture qualitative behavior and not exact quantiative values, it follows that the ensemble members vary slightly on the edge of the domain. All of the ensemble members merge together by roughly 20 mm until almost the end of the grid. This indicates an approximately 0 spread, meaning there is high certainty in the wave dynamics.

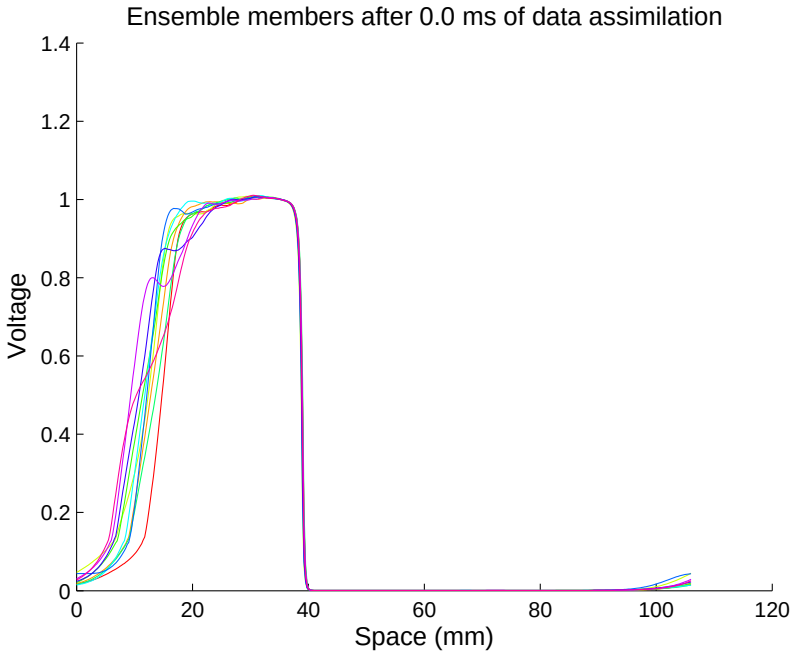


Figure 10: Beginning u ensemble obtained from breeding.

The algorithm also requires a function, $\mathbf{H}$, which can be linear or non-linear. $\mathbf{H}$ is the map between the $\mathbf{x}^{b(i)}$'s and the $\mathbf{y}^{b(i)}$'s, the ensemble members in the observation space. Let ℓ be the number of observations used in our domain, over all three variables. Then, each $\mathbf{y}^{b(i)}$ is $\ell \times 1$. We assume $\mathbf{H}$ to be a linear map since we want to replicate real-world experiments, which approximately have evenly spaced time and spatial observations. Since we can either have observations at some or every point in the model space, we must have $\ell \leq m$. Multiple observations could be used at the same grid point, which could cause $\ell > m$, but we assume there is at most one observation at each point. We also provide the algorithm with an ℓ-dimension column vector, $\mathbf{y}^o$, of observations. In our case this is the control state variables, u, v and w, appended together in the same manner as the $\mathbf{x}^{b(i)}$'s. Finally, LETKF requires an $\ell \times \ell$ observation error covariance matrix, $\mathbf{R}$. We use a diagonal matrix for $\mathbf{R}$ where each entry is 0.05 or the standard deviation of the Gaussian curve where our perturbation values come from. By adding perturbation values to the variables from the Gaussian curve mentioned before during the breeding process, we can assume that the majority of our ensemble members should be within one standard deviation of the control state. The output of LETKF is k $m \times 1$ vectors, called the analysis ensemble, denoted by $\mathbf{x}^{a(i)} (i = 1, 2, \ldots k)$. For the remaining iterations of data assimilation, we run the resulting analysis ensemble through the FK model and use those as the new background ensemble,

which gives the analysis ensemble model-dependent error while progressing the analysis ensemble to the next time step for data assimilation.

Also important to note is ρ, the multiplicative covariance inflation factor. The algorithm on its own tends to underestimate the uncertainty in the initial guess. This equates to an overconfidence in the background ensemble, and thus the filter puts less of a weight on the observations. If the discrepancy between the background ensemble and the observations becomes too large, the observations end up being ignored when building the analysis ensemble. ρ helps put more weight back onto the observations by contributing to the spread of the ensemble members, or how far apart they are from each other. The larger ρ is, the larger the spread of our ensemble should be. A value of $\rho = 1$ will not change the spread, $\rho < 1$ will decrease the spread, and $\rho > 1$ will increase it [24].

We are now able to list the steps of the LETKF algorithm:

5.2.2 The Algorithm

1. Map the $\mathbf{x}^{b(i)}$'s to the $\mathbf{y}^{b(i)}$'s using the $\mathbf{H}$ matrix The superscript b stands for background.

   Find the average of all the $\mathbf{y}^{b(i)}$, call it $\bar{\mathbf{y}}^b$, and subtract each $\mathbf{y}^{b(i)}$ from $\bar{\mathbf{y}}^b$ to form the columns of the $\ell \times k$ matrix, $\mathbf{Y}^b$.

2. The same process is done with all of the $\mathbf{x}^{b(i)}$'s to find $\bar{\mathbf{x}}^b$ and the $m \times k$ matrix $\mathbf{X}^b$, or background ensemble.

3. This step is the localization, which is skipped for the reasons given above.

4. Compute the $k \times \ell$ matrix $\mathbf{C} = \left(\mathbf{Y}^b\right)^{\mathbf{T}} \mathbf{R}^{-1}$. We achieve this by solving $\mathbf{RC^T} = \mathbf{Y}^b$ to avoid inverting matrices. $\mathbf{C}$ has no particular meaning; calculating it is an intermediate step for ease of computation and implementation.

5. Compute the $k \times k$ covariance matrix $\tilde{\mathbf{P}}^a = \left[(k-1)\,\mathbf{I}/\rho + \mathbf{CY}^b\right]^{-1}$. The superscript a stands for analysis and $\rho >$ is the multiplicative covariance inflation factor.

6. Calculate $\mathbf{W}^a = \left[(k-1)\tilde{\mathbf{P}}^a\right]^{1/2}$. W weights the spatial locations of the background ensemble members in $\mathbf{X}^b$, when adding the members to the background estimate, $\bar{\mathbf{x}}^b$ in step 8. $\mathbf{W}^a$ is calculated by using the eigenvalues and eigenvectors of $\tilde{\mathbf{P}}^a$ for computational efficiency.

7. Compute the k-dimensional vector $\bar{\mathbf{w}}^a = \tilde{\mathbf{P}}^a\mathbf{C}\left(\mathbf{y}^o - \bar{\mathbf{y}}^b\right)$. Then, add $\bar{\mathbf{w}}^a$ to each column of $\mathbf{W}^a$.

8. Multiply $\mathbf{X}^b$ by each $\mathbf{w}^{a(i)}$ and add $\bar{\mathbf{x}}^b$ to get the analysis ensemble members $\mathbf{x}^{a(i)}$.

9. After performing Steps 3 — 8 for each grid point, the outputs of Step 8 form the global analysis ensemble. This step is skipped since we are not using the localization step [20].

The algorithm works mathematically by minimizing the cost function,

$$\tilde{J}^* (\mathbf{w}) = (k-1)\, \mathbf{w}^T \mathbf{w} + \left[\mathbf{y}^o - \bar{\mathbf{y}}^b - \mathbf{Y}^b \mathbf{w}\right]^T \mathbf{R}^{-1} \left[\mathbf{y}^o - \bar{\mathbf{y}}^b - \mathbf{Y}^b \mathbf{w}\right], \tag{5.1}$$

by solving

$$\bar{\mathbf{w}}^a = \tilde{\mathbf{P}}^a \left(\mathbf{Y}^b\right)^T \mathbf{R}^{-1} \left(\mathbf{y}^o - \bar{\mathbf{y}}^b\right), \tag{5.2}$$

$$\tilde{\mathbf{P}}^a = \left[(k-1)\, \mathbf{I} + \left(\mathbf{Y}^b\right)^T \mathbf{R}^{-1} \mathbf{Y}^b\right]^{-1}, \tag{5.3}$$

and

$$\bar{\mathbf{x}}^a = \bar{\mathbf{x}}^b + \mathbf{X}^b \bar{\mathbf{w}}^a. \tag{5.4}$$

6 Results from Data Assimilation

This section of the thesis begins by discussing how results from data assimilation will be analyzed. Results from the most basic simulation for data assimilation in this context are given and evaluated to raise confidence in our adaption of the ETKF algorithm. To further increase our confidence, observations are removed from the system, giving ETKF a more difficult estimation problem. Once the implementation of ETKF has been verified, synthetic observations more similar to real-world experiments are used and initial conditions for data assimilation are modified. Time series of the data assimilation results are compared with a time series of results from the FK model alone. The final portion of this section discusses parameter estimation results.

6.1 Methods of Evaluation

The goal of data assimilation is to approximate given observations. After building our data-assimilation dynamical system based on the FK model, we run experiments with different observations of our u, v, and w variables. By using observations at every grid point, we get an idea of the best-case scenario, while using a small number of observations can help us test the limits of the algorithm. We can also match the number of observations to experimental conditions. We primarily change the observations in each simulation by changing $\mathbf{H}$. Different maps for $\mathbf{H}$ can allow us to use the same observations, but we can map them to different spaces. For example, our final grid size is 425. If we use the 3*425$\times$3*425 identity matrix for $\mathbf{H}$ on our observation, we will have values on every grid point for each variable in our observation space. This map, however, is not entirely practical experimentally. It does, however, show us how well data assimilation can do in a best-case scenario, as mentioned above. We may only be able to see observations at every fifth grid, on the first half of the grid, just the middle of the grid and so on. Cameras used in these real-life experiments take observations every approximately every 2 ms at every 0.031-0.047 cm and only for the u variable. For our work, by assuming we only have observations for the u variable at every 25 grid points, or 0.0625 cm, every 50 ms, we have a much more realistic setting for the observations, yet with slightly worse conditions than real experiments. By modifying $\mathbf{H}$, we can create these real, experimental possibilities. We can also modify the observations by adding error to them and combine this with changing the $\mathbf{H}$ map.

In order to test the robustness of this dynamical system, we have also run simulations that involve

creating an initial background ensemble containing members from different model times from the breeding experiments. In this case, each member still has dynamics based on the FK model, but by averaging these ensemble members together, we create a very impractical first guess, which we can use to see if the algorithm can eventually create an ensemble that matches the given observations [25]. Cardiac modeling has typically been focused on trying to recreate qualitative behavior seen in experiments, and not necessarily trying to exactly match observed data. Because of this, simulations to replicate quantitative results do not seem to have an accepted initialization method in this area of study.

To demonstrate that data assimilation is working properly, the analysis ensemble is verified to be a better approximation to the given observations than the background ensemble. This can be done in a variety ways. The main way uses the average of each ensemble. The average of the analysis ensemble is the analysis estimate and the average of the background ensemble is the background estimate. We compare the background and analysis estimates with the known truth over time, or in our case, the numerical solution to the FK model that we have taken to be the truth. The next way involves plotting both ensemble averages with our observation. Here, we can visually see if the analysis estimate is closer to the observed values than the background estimate. This method is not preferred, since it does not give an exact comparison, but it does show FK model dynamics for the background and analysis estimates as well as the observations. Another method involves measuring the RMS between each ensemble average and the observed values. Again, the RMS for the analysis ensemble should be lower than the background ensemble. This method is similar to the previous method, but standardizes the distances from each other through the use of the RMS norm and removes the dynamics of the system at each particular time for a more concrete comparison. This tactic is also similar to the first one mentioned, but varies slightly because the set of values we take as the observations is generally different from the set taken as the truth. One more method uses the spread of each ensemble across every grid point. The spread is a measure of how far apart the members are from each other. A way to measure this is by calculating the RMS of the difference between each ensemble member and the mean of the ensemble. Since the analysis ensemble should be a better approximation than the background ensemble, the spread should be lower for the analysis ensemble when compared to the background ensemble.

While it is desirable to see that the analysis from data assimilation is an improvement over the background guess, another comparison to make is between the analysis estimate and the FK model alone.

By doing so, the benefits of data assimilation with the FK model can be seen directly. To do this, an initial background ensemble average was run through the FK model for 3000 ms, the same duration that ETKF is run for. This is called the free run. The error between the truth and the free run, the background guesses and the analysis output from ETKF were then found respectively using the RMS norm. If data assimilation is doing better than FK alone, then the error between the truth and the free run should be higher than the RMS between the truth and the background guess which should be higher than the RMS between the analysis and the truth.

From this point on, a time of 0 ms will mean no elapsed time for data assimilation, and correspond to a time of 6100 ms of FK model time. In addition, due to the low growth rates and similar growth patterns seen in the breeding experiments over different breeding intervals, no variation in time intervals for data assimilation is necessary. The only time interval we use for data assimilation is 50 ms. This time interval is longer than the interval in which real experimental observations can be taken. So, we know that if data assimilation does well with this time interval length, it will do better with a shorter one. This longer time interval also helps maintain computational efficiency and data storage, and since low growth rates were seen during breeding, numerical accuracy will not be compromised much.

All of the following simulations use a ρ of 1 unless otherwise noted. $\rho = 1$ will not effect the spread of the background ensemble members. An ensemble size of 10 is used for the first few simulations and is varied after. Idealized realistic observations will correspond to using observations every 25 grid points, for u only, while idealized, almost-realistic observations will correspond to observations still at every 25 grid points, but for all three variables. Later, error will be added to the observations from a Gaussian curve with mean 0 and standard deviation of 0.05 and these sets of observations will no longer be referred to as idealized.

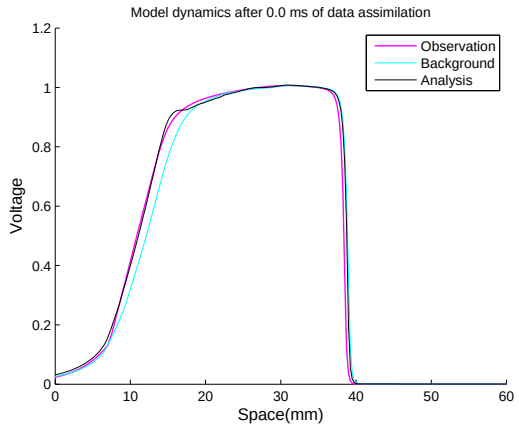
6.2 Initial Simulations

Here, observations at every grid point for all variables, idealized, almost-realistic observations, and observations at every 85 grid points for just the u variable are compared to ensure an accurate implementation of data assimilation with the FK model. We expect the simulations with more observations to perform better than those with less observations. Because the last observation distribution with observations only at every 85 grid points for just the u variable is so extreme, we do not expect the analysis estimate to be an improvement over the background estimate. For each of these simulations

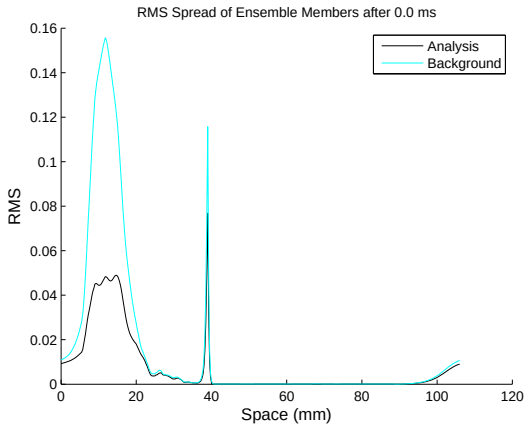
in the initial simulations, the control variables were used as observations.

For the first simulation, the control variable solutions were used as observations and **H** was the 3*425×3*425 identity matrix. By using this choice for **H**, it is assumed that there is an observation available at every grid point, for all three variables. This is not a realistic set of observations since we cannot observe the v or w variables, and spatially, we cannot have observations at every grid point for this grid step size. This observation distribution can verify that the algorithm has been properly adapted and implemented. This is the easiest case for data assimilation to provide an improved analysis. If the implementation fails to provide a better guess than the background, then there is an error in the code to fix.

Figure 11 points to this implementation of ETKF being correct. This figure only shows the first iteration of ETKF. Similar results hold for other times and for the v and w variables as well. One thing to note in Figure 11b is that for the grid points approximately between 20 and 38 mm, the spread is almost 0. This could be good or bad. On one hand, all of the ensemble members agree on those grid points, but on the other, if all of the ensemble members do not agree with the observations, then the system is suffering from ensemble collapse, which will be discussed more below. The analysis estimate is closer to the observation than the background estimate and the spread for the analysis ensemble members is lower than background in most places. Again, this is the case where we assume we have an observation at every grid point, and since the results of this simulation are positive, more realistic simulations can be run with higher confidence.



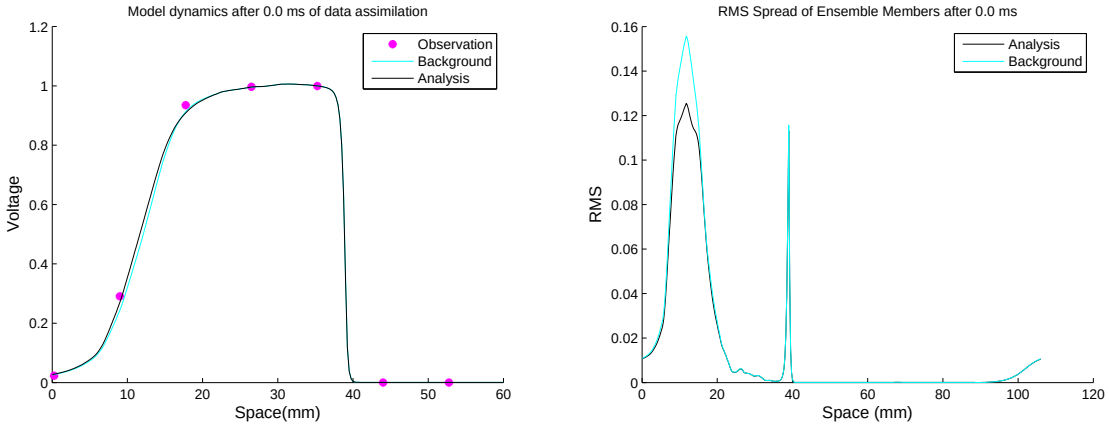
(a) Dynamics from data assimilation.



(b) RMS spread between the background guess and analysis for u.

Figure 11: The observation (at every grid point), the background estimate, and the analysis estimate for u after 1 iteration.

Next, the same experiment as above was conducted, except with idealized, almost-realistic observations. Figure 12 below shows results from the first iteration of data assimilation from this experiment. The background estimate is the same as above since it is the same initial ensemble, and this is the first iteration of ETKF. The analysis estimate, however, is closer to the background estimate when compared to using observations at every grid point as we would expect since there are less observations. We also see a greater spread when compared to above (see Figure 12b) in the analysis ensemble when the observations are idealized almost-realistic than when there are observations at every grid point.

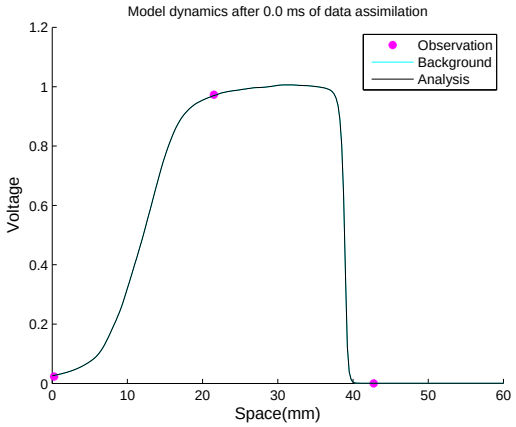


(a) Dynamics from data assimilation.

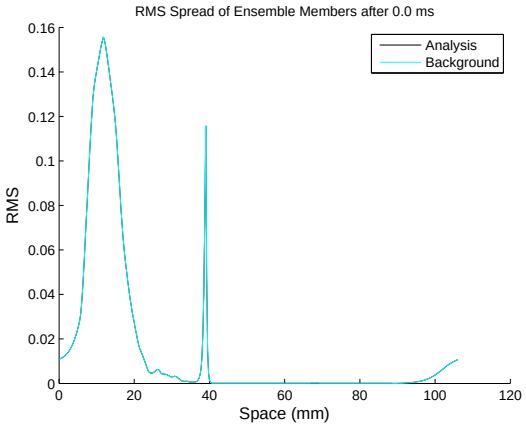
(b) RMS spread between the background guess and analysis for u.

Figure 12: Idealized, almost-realistic observations, the background estimate, and the analysis estimate for u after 1 iteration.

To test the limits of the ETKF algorithm, an even more dramatic case is using only 5 observations spaced 85 grid points away from each other, and using only u in our observations. We can see in Figure 13 that the background estimate and the analysis estimate are almost on top of each other and their spreads are also roughly the same. Without more observations, ETKF cannot get the analysis estimate any closer to the observed values. This case shows us that we need more than five observation points on the grid to have dramatically improved estimates; here the analysis is only a slight improvement to the background.



(a) Dynamics from data assimilation.



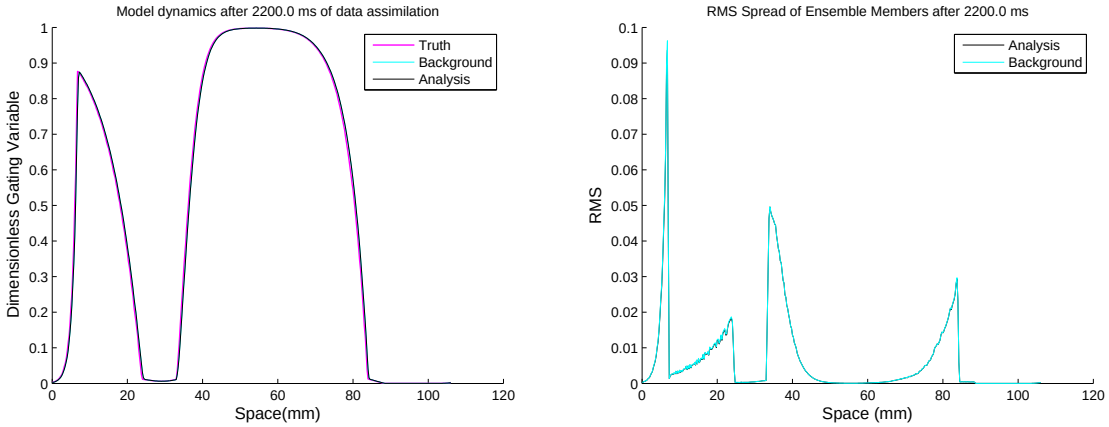
(b) RMS spread between the background guess and analysis for u.

Figure 13: The observation (at every 85 grid points) for just the u variable, the background estimate, and the analysis estimate for u after 1 iteration.

6.3 Idealized Observation Experiments

The only observation distributions discussed in this section are the idealized, realistic observations and the idealized, almost-realistic observations. For these observation distributions, we use both good initial conditions (ensemble members all from time 0 ms) and random initial conditions which are from random breeding intervals. Individual experiments are covered first, followed by time series comparisons.

We begin with idealized, realistic observations and good initial conditions for the background ensemble. This is a much more practical scenario since neither v nor w can be observed experimentally. The plots for u in this case look very similar to Figure 12, when we used idealized, almost-realistic observations. Below are figures of v from the simulation with observations from all three variables. Similar figures were generated with the same settings, except the removal of observations from v and w. The same is true of the w variable. Since only u can be observed experimentally, this is a very good sign for the ETKF algorithm, since similar results can be achieved using only observations from u versus observations from all three variables.



(a) Dynamics from data assimilation.

(b) RMS spread between the background guess and analysis for v.

Figure 14: The "truth", the background estimate, and the analysis estimate for v after 2200 ms. for idealized, realistic observations.

Now that we know data assimilation can preform well with good initial conditions and idealized, realistic observations, the next simulations presented use ensemble members from different time steps, creating random initial conditions paired with the idealized, almost-realistic and idealized realistic observations. By doing so, each member is a reasonable first guess, but the average of all the members is not a reasonable guess [25]. Because each member individual displays model characteristics, each can be integrated through the FK model after it is used in the ETKF algorithm, whereas completely random guesses most likely would cause wave annihilation in the model. Figure 15 is a spaghetti plot of these initial background ensemble members for the u variable.

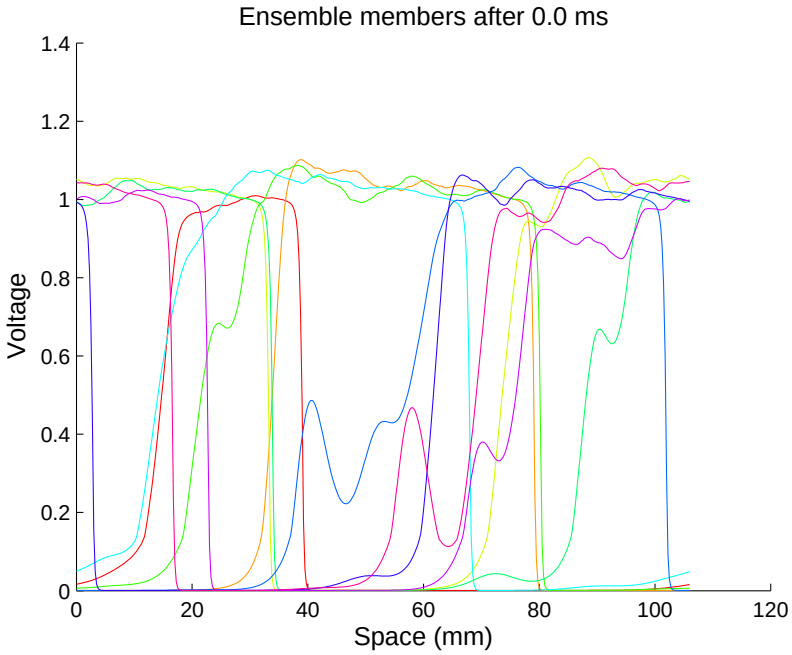
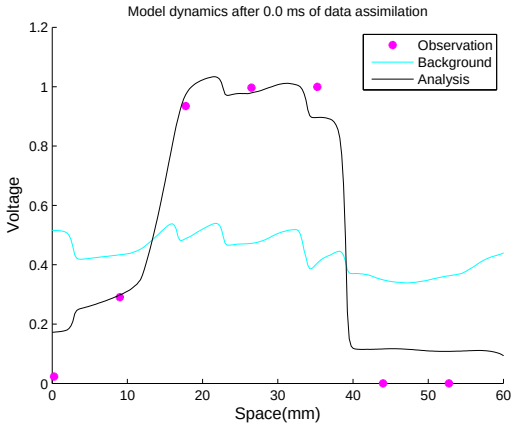
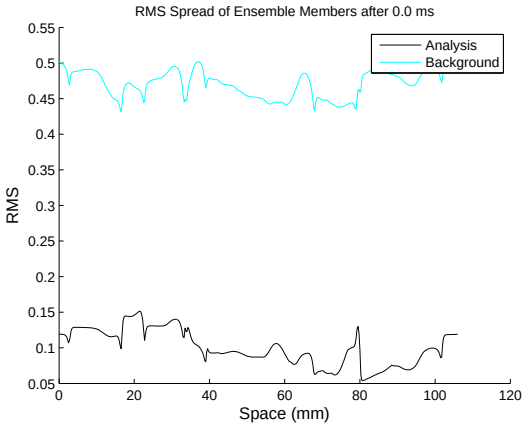


Figure 15: Random initial ensemble members for u from breeding.

We can see in Figure 16 that the initial background estimate is not very good, but Figure 15 shows us that each ensemble member has the dynamics of u. The resulting analysis estimate using idealized, almost-realistic observations after one iteration of ETKF is much better than the random initial conditions and begins to have some features similar to the truth. The spread of the analysis ensemble is much lower than the first guess, as expected.



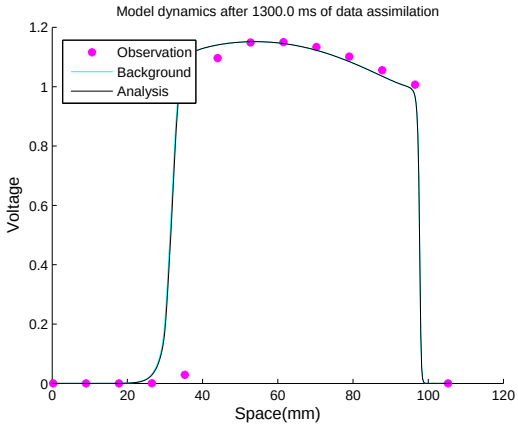
(a) Dynamics from data assimilation.



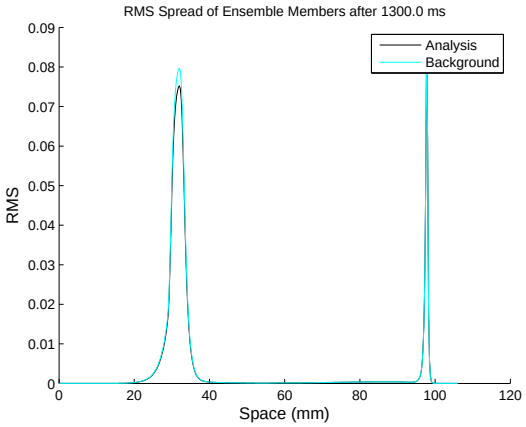
(b) RMS spread between the background guess and analysis for u.

Figure 16: Idealized, almost-realistic observations, the background estimate, and the analysis estimate for u after 1 iteration.

After 500 ms, the ensemble members begin to converge close together. By 1300 ms, the spread of the background ensemble is about 20% of the spread it started with at 100 ms. Between 0 and 20 mm, and 40 to roughly 85 mm, the spread of both ensembles is approximately 0, meaning the members are in agreement on their spatial locations. Both ensembles now resemble the observation. Figure 17 is 1300 ms after the initialization. It shows that there is a slight spatial lag in the ensemble compared to the observed values.



(a) Dynamics from data assimilation.

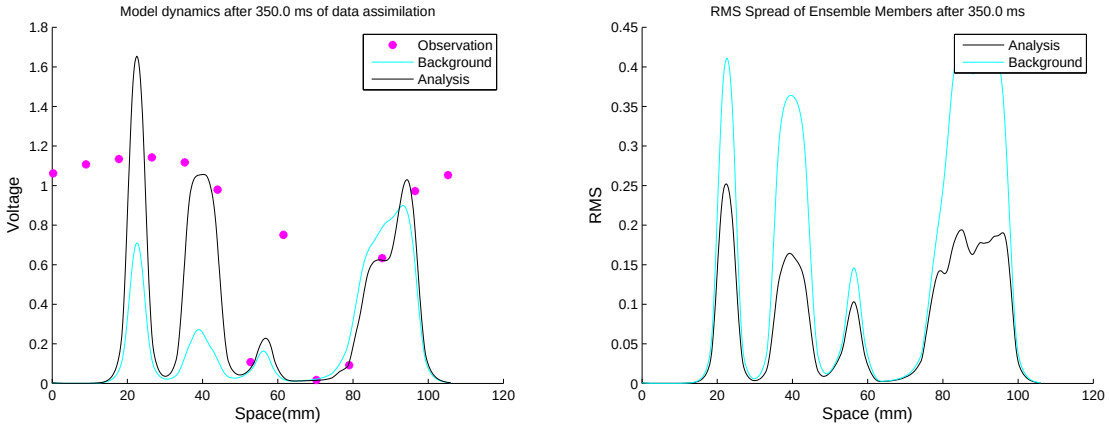


(b) RMS spread between the background guess and analysis for u.

Figure 17: Idealized, almost-realistic observations, the background estimate, and the analysis estimate for u after 1300 ms.

The spatial lag is an artifact of the dynamical system we have created. Since the spread is approximately 0 for a majority of the domain, the ensemble members thus all agree in voltage. So, there will not be enough weight on the observed values to correct the analysis estimate. This spatial lag is known as ensemble collapse, and will be discussed in the next section.

If we run the same simulation as above except using idealized realistic observations, the results from the first few iterations look similar to the experiment using idealized, almost-realistic observations. But when the simulation has run for 350 ms, the ensemble members of u begin to tend toward 0 and the members of v and w begin to tend toward 1. By 500 ms, all ensemble members of u are approximately 0 (within $\times 10^{-4}$) and the members of v and w are all approximately 1. When the variables tend toward these values, this tells us that the system is collapsing (leading to wave annihilation). We cannot prevent this from happening in this case because of the low spread in the ensembles. All of the members are over-confident in state variable values differing from the used observations, so the observed values are not enough to decrease the error between the ensemble estimates and observations.



(a) Dynamics from data assimilation.

(b) RMS spread between the background guess and analysis for u.

Figure 18: Idealized, realistic observations, the background estimate, and the analysis estimate for u after 350 ms.

The results mentioned thus far primarily compare background and analysis ensembles. Another aspect to check, as mentioned at the beginning of this section, is if data assimilation is doing a better job than just the FK model would on its own. Here, we compare the error between the free run and analysis with what we have taken to be the truth over time.

In many data assimilation experiments, an ensemble size between 20 an 100 is typically used. Since

we did not have success with only 10 ensemble members for the above scenario with idealized, realistic observations, we will increase the ensemble size to 20 random members. In doing so, data assimilation is successful in creating improved estimates even with only observations from u, which we just saw fail with only 10 ensemble members. After some high initial error (see Figure 19), the RMS of the background and analysis drop below the RMS of the free run of a non-random ensemble average. This tells us that even with a very unrealistic guess and lower spatial resolution than a real experiment, data assimilation has surpassed results from the FK model alone, which started from good initial conditions.

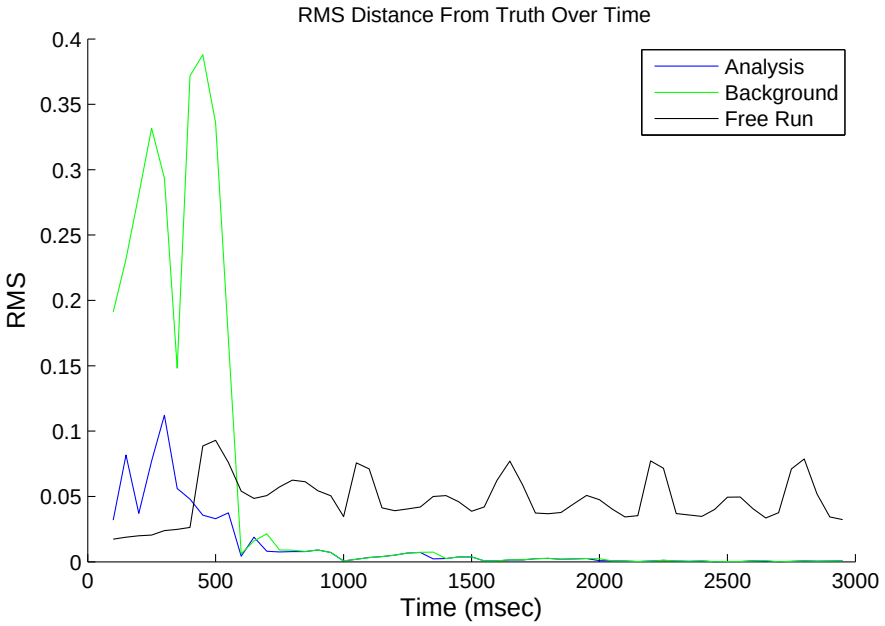


Figure 19: RMS error over time using a random initial ensemble with 20 members and idealized, realistic observations compared with a non-random ensemble average free run.

To determine which settings for data assimilation are optimal, each simulation needs to be compared to one another as well as with the FK model alone. This method is the best for evaluating our results since it standardizes the distances from each other and removes the dynamics of the system at each particular time through the use of the RMS norm. The RMS between each ensemble average and the observed values will be measured. If data assimilation is doing better than the free run, the RMS for the analysis estimate should be lower than the RMS for the FK free run.

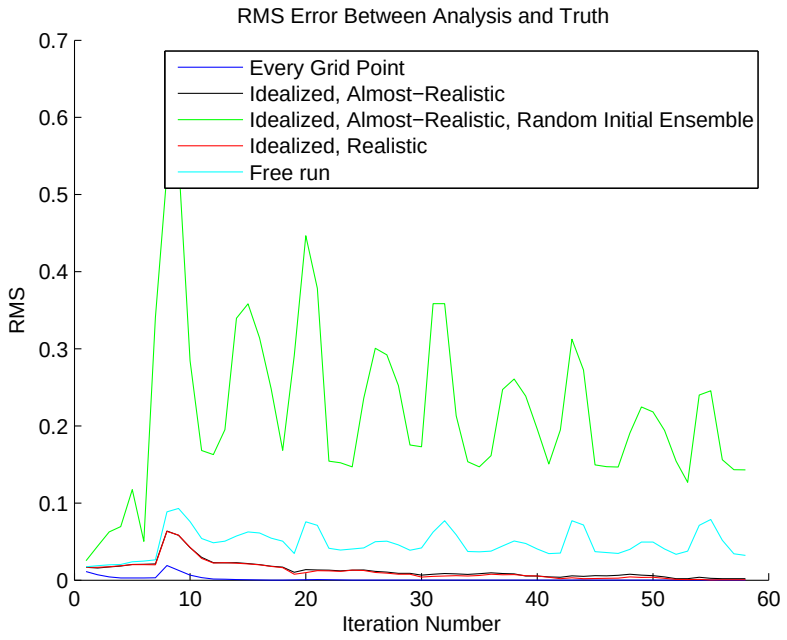


Figure 20: Error in the analysis ensemble of size 10 from the truth over time for u with different observation distributions and initial ensembles compared with a non-random ensemble average free run.

Figure 20 shows the time series of the analysis estimate for u with different observation distributions and a non-random ensemble average free run. It is consistent with our results presented earlier in this section. Using an observation at every grid point, while unrealistic, did the best job by minimizing the RMS distance between the observation and data assimilation analysis. Using idealized, almost realistic observations did as well as idealized realistic observations with good initial condtions, but performed better than when using idealized, almost realistic observations with random initial ensemble members. When using random ensemble members we can see that the FK model run does better. In all four situations, the RMS distance for the background and analysis does decrease slowly, but linearly, over time.

The most realistic simulation we have described so far is is the idealized realistic set of observations with values at every 25 grid points only from u, since this resolution is lower than in real experiments and the only variable in this model that we can actually observe is u. Finding a balance between accurate analyses from data assimilation and computation time is important. Being able to find a threshold where increasing the number of ensemble members does not add much accuracy is a good way to save computation time for our model.

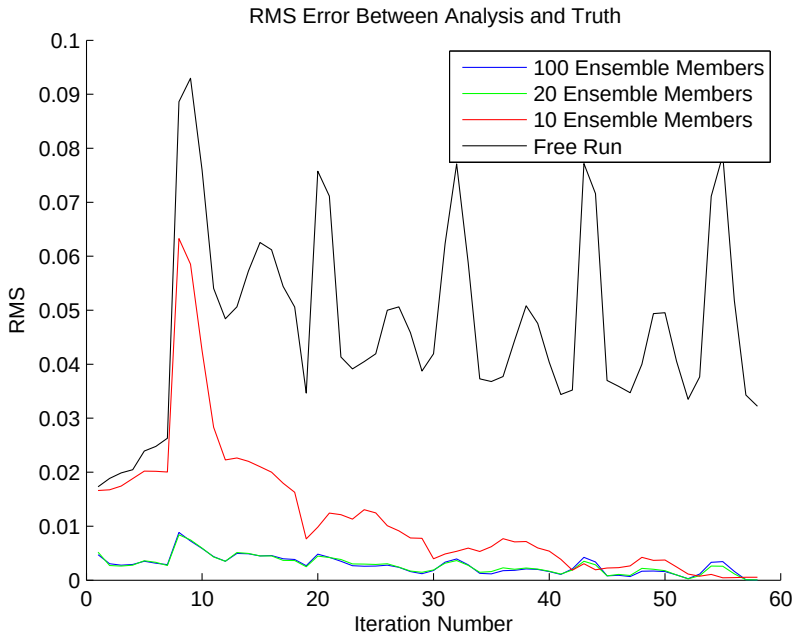


Figure 21: Error in the analysis estimate for u with idealized, realistic observations with different ensemble sizes compared with a non-random ensemble average free run.

Figure 21 shows the time series of the analysis estimate for u with varying ensemble sizes with a non-random ensemble average free run. The free run again has the highest error from the truth. First note that the order of magnitude of error in Figure 21 is roughly 10% of the order of magnitude in Figure 20. We also see, in Figure 21, that for the 10 member ensemble, the error spikes initially, then decreases linearly, until the error is about same as the 20 and 100 member ensembles after 42 iterations. Between 50 and 60 iterations, the 10 member ensemble seems to out perform the other ensembles, but this is most likely due to ensemble collapse. The 20 and 100 member ensembles have a similar, but smaller spike at first, but have approximately one-sixth or 16.67% of the error that the 10 member ensemble has during this time. We can see also that the error for the ensemble sizes of 20 and 100 are approximately the same throughout, and for some iterations, the error is actually slightly lower for 20 members. Thus, we can conclude that without sacrificing much accuracy, using 20 ensemble members seems to be the optimum ensemble size when were are not focused on correcting ensemble collapse.

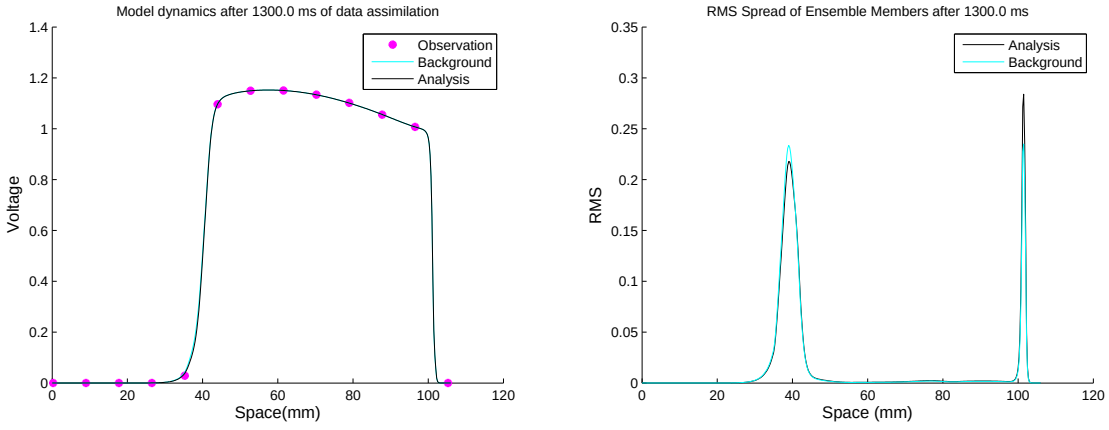
Due to the lack of chaos seen in the FK model, it follows that increasing the ensemble size beyond 20 does not contribute much to decreasing the error from the truth. With a low number of unstable subspaces in our system, the ensemble size of 20 is large enough to make corrections in all or most of

the subspaces.

6.4 Ensemble Collapse

When the ensemble members all converge together and have a spread of approximately 0, the situation is referred to ensemble collapse. It is a known issue with the ETKF algorithm [22]. In this context of this cardiac, data-assimilation problem, when all or most of the ensemble members converge together and exhibit the same wave dynamics in space and time, this is known as ensemble collapse. This section of the results discusses the problem caused by ensemble collapse, and how to correct it. Individual simulation results are given, then followed by a time series comparison.

There are a few ways to deal with ensemble collapse. The first is simply increasing ρ, the multiplicative covariance inflation factor, since it puts a heavier weight on the observations by increasing the spread of the background ensemble during the data-assimilation process. This is a common strategy since it is computationally efficient and it is usually necessary due to a small ensemble size. In the results mentioned so far, a value of $\rho = 1$ has been used. After raising this value to 2, we have concluded that $\rho = 2$ is enough to correct the spatial lag. This simulation used idealized, realistic observations, good initial background ensemble members and an ensemble size of 10. This correction in the spatial lag is seen in the wave dynamics of Figure 22a. The observations in are in agreement with the analysis and background estimates, unlike what we saw in Figure 17.



(a) Dynamics from data assimilation.

(b) RMS spread between the background guess and analysis for u.

Figure 22: The idealized, realistic observations, the background estimate, and the analysis estimate for u after 1300 ms with good initial background ensemble members and an ensemble size of 10 and $\rho = 2$.

To test the robustness of the system, ran the same simulation as in Figure 22 above, but now with $\rho = 4$. The wave dynamics seen in the analysis and background ensembles suggest that this choice of ρ is too large since it gave rise to local, spatial behavior, which has not seen in any model-generated truth or observed value.(see Figure 23).

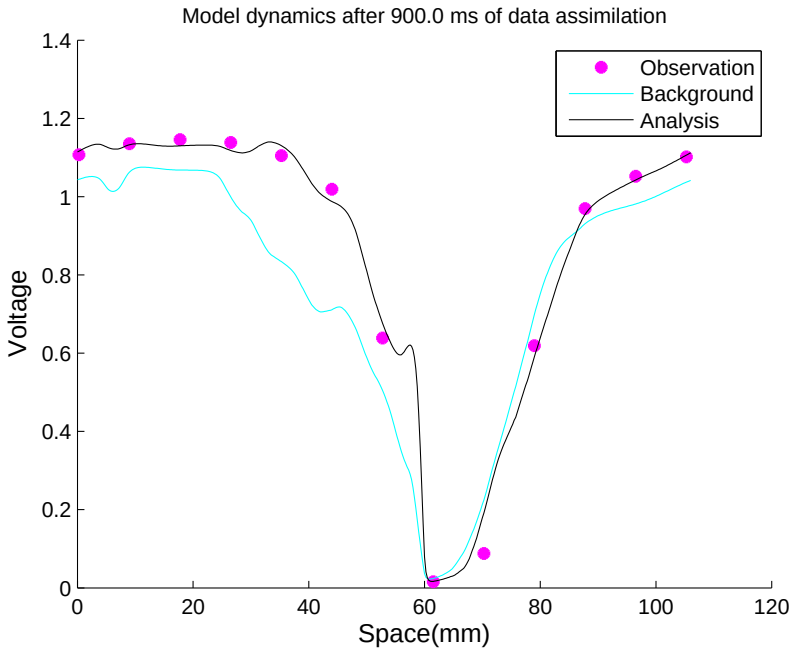
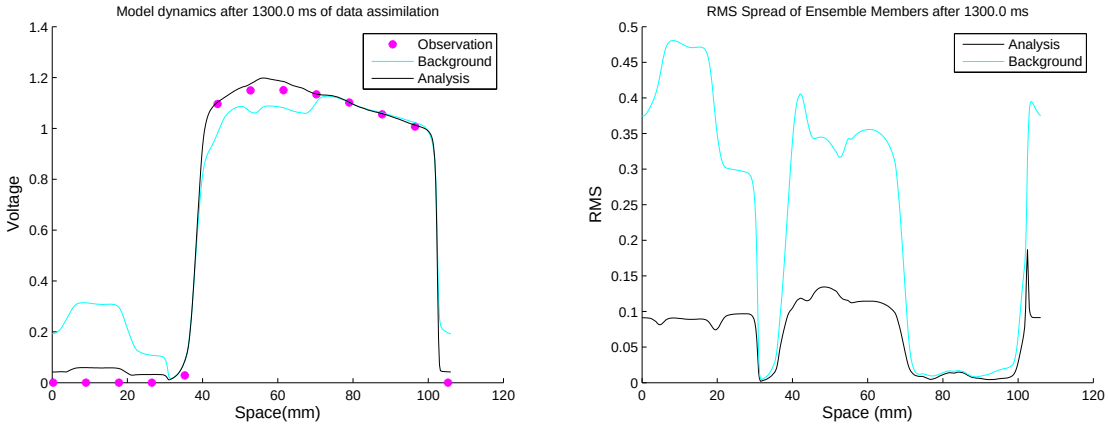


Figure 23: The idealized, realistic observations, the background estimate, and the analysis estimate for u after 900 ms with $\rho = 4$.

The second method for correct ensemble collapse involves adding additional perturbations to each of the ensemble members to increase their spread constantly. In a data-assimilation experiment conducted by Hamill and Whitaker, perturbations were added in this manner every 6 hours in their 72 hour simulation, or 8.33% of the time [23]. This suggests that for our simulations, we should introduce these extra perturbations approximately every 250 ms in the same manner as the error analysis. We add values from a Gaussian distribution of mean 0 and standard deviation of 0.05 to every grid point, for all variables and all ensemble members. We can see in Figure 24b that there are some areas of the grid with high certainty, or spread close to 0, which is good. On the other hand, in Figure 24b, there are other portions of the domain where the background ensemble does not look like a good initial guess since its average is not very smooth like in other simulations at this elapsed time. This tells us that analysis ensembles with perturbations every 250 ms did not converge together very well before

the next perturbation when using only 10 ensemble members.

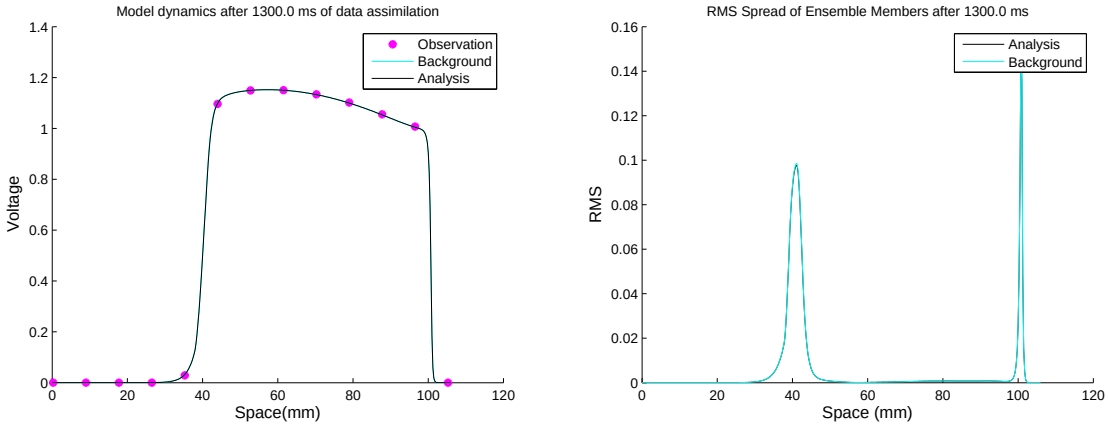


(a) Dynamics from data assimilation.

(b) RMS spread between the background guess and analysis for u.

Figure 24: The idealized, almost-realistic observations, the background estimate, and the analysis estimate for u after 1300 ms.

Figure 24 above is a simulation with 10 ensemble members. If we increase the number of members to 20, Figure 25 shows that data assimilation gets a much better approximation than with an ensemble size of 10. The local behavior seen in the simulation with 10 ensemble members is now gone and the spatial lag caused by ensemble collapse has been corrected since the observed values now have approximately the same voltage as the background and analysis estimates.



(a) Dynamics from data assimilation.

(b) RMS spread between the background guess and analysis for u.

Figure 25: Idealized, realistic observations, the background estimate, and the analysis estimate for u after 1300 ms with 20 ensemble members and additive inflation every 250 ms.

We will now compare of all the methods seen in this section against each other and with a free run. Figure 26 below shows the time series of the analysis estimate for u with the different ensemble collapse treatments. The order of magnitude in this time series is about half of what we saw in Figure 20 and five times the magnitude of Figure 21. The superior method in correcting the ensemble collapse is using $\rho = 2$. It consistently has the lowest error for each iteration of data assimilation. The other methods all interchange each iteration as to which has the lowest error. Using $\rho = 4$ seems to be the most unstable since its error spikes for the last few iterations. The advantage to $\rho = 2$ proving to be the best at correcting ensemble collapse is that it is the computationally more efficient than the additive inflation method.

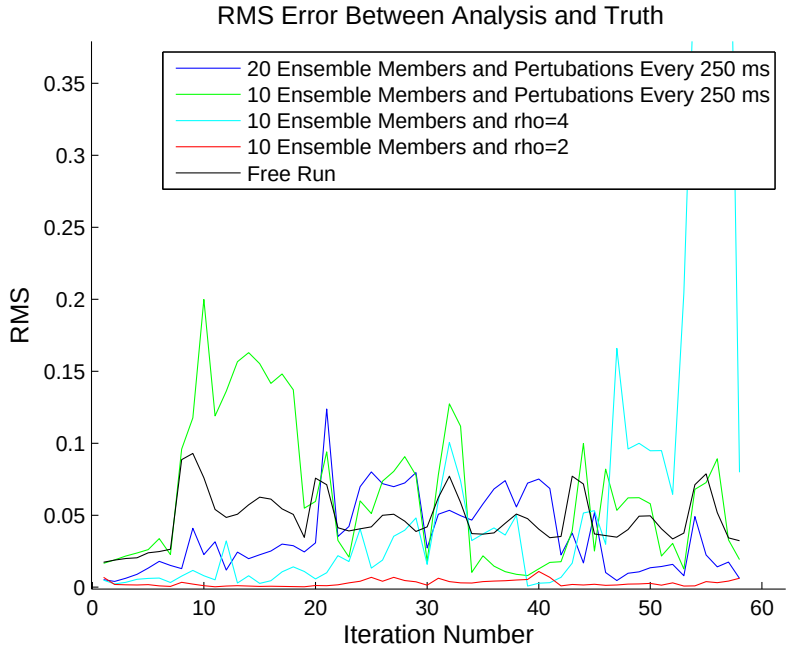


Figure 26: Error in the analysis ensemble from the truth over time for u with idealized, realistic observations with different methods to correct ensemble collapse compared with a non-random ensemble average free run.

6.5 Realistic Observation Experiments

Up to this point, only observations without error have been discussed. By taking the observations we have used up to this point and adding error from a Gaussian distribution with mean 0 and standard deviation of 0.05, the same used to make pertubations to the variables, we can see how robust the system is by seeing if ETKF can maintain a reasonable analysis estimate.

Below in Figure 27, idealized, realistic observations were perturbed, making them realistic observations. The free run was not included in Figure 27 since its magnitude is roughly 10 times larger than the include scenarios. Non-random initial conditions were used, the ensemble size was varied. For all three ensemble sizes, the error decreases linearly after the initial start up. The error in this case is actually higher for the larger ensemble size. This is most likely due to ensemble collapse. The ensemble members all agree on a certain voltage, while the observations disagree more than they would if the observations were ideal. The more ensemble members there are, the more weight would be needed move the ensemble members to the realistic observations. Using random initial background ensembles led to wave annihilation for the background and analysis estimates and ultimately terminated after 5 iterations or less of data assimilation.

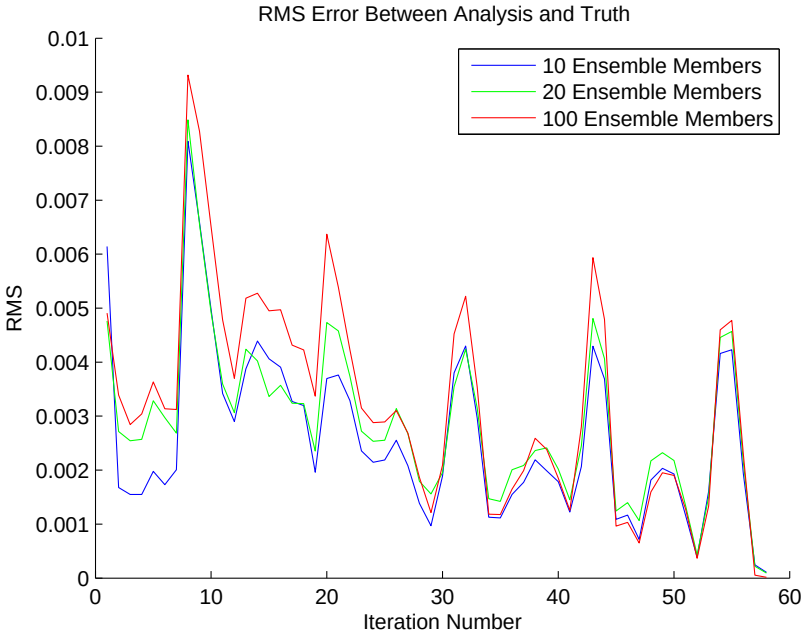


Figure 27: Error in the analysis ensemble from the truth over time for u for realistic observations with varying ensemble sizes.

6.6 Parameter Estimation

The final set of simulations involves the parameter-estimation application. Perturbed parameter values were appended to the end of the background ensemble members to be a part of the initial conditions for data assimilation. Perturbations came from Gaussian curves with standard deviations that were within physical meaning for each parameter. For example, τ_d is only 0.25. Perturbing it to 0.5 (doubling it),

will not make much physical sense, whereas τ_w^+ could handle a perturbation of 0.25 from above since its value, 870, is so large. Because there are not exact values for these that should be incorporated, no observations for the parameters are used.

When choosing parameters to estimate, threshold values were not selected, as they are used in the Heaviside step function to determine which parts of the equations are used. In initial simulations, τ_r, τ_o, τ_{si}, τ_w^+, and τ_w^- were chosen and used together in the initial background ensembles. The perturbed parameters in each ensemble member all converged together after only 2-3 iterations. However, the change in the parameter values caused the variables not to resemble the observed values at each time step for various standard deviations. Idealized, realistic observations were used in these simulations.

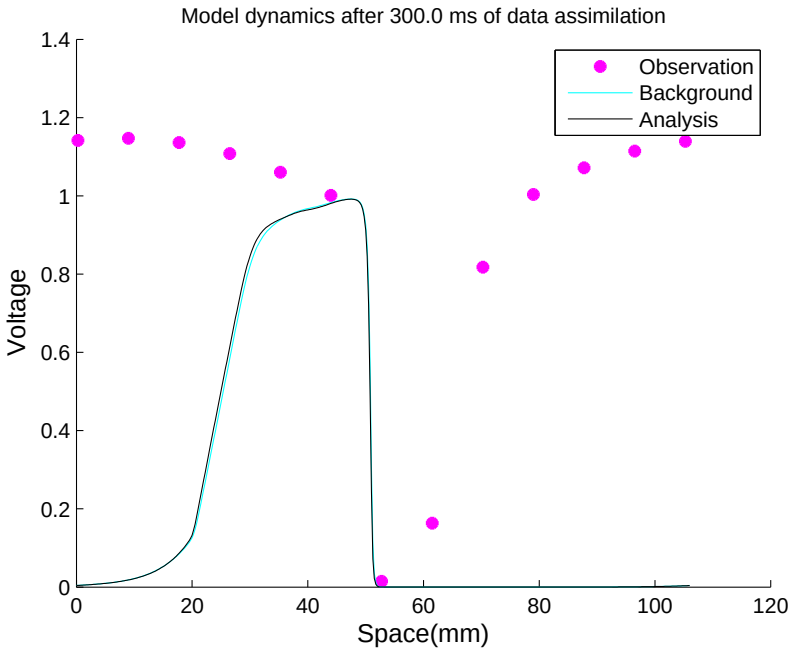


Figure 28: Wave dynamics from data assimilation for the u variable with parameter estimation for τ_r, τ_o, τ_{si}, τ_w^+, and τ_w^-.

Figure 28 is an example of how far off the analysis and background ensembles end up from the observed values. The analysis and background do, however, still maintain wave dynamics. Since using five parameters seemed to be too many, one parameter was used at a time from this group for parameter estimation. Results similar to Figure 28 were seen. Simulations estimating τ_d, τ_{v1}^-, and τ_{v2}^- were also done separately. All of these experiments also produced similar results.

7 Conclusions and Future Work

To the best of our knowledge, this is the first approach combining data assimilation techniques with a cardiac cellular model. This approach was used because of a lack of data from observational methods for electrical waves in the heart and limitations with numerical models. Because of the coupling of the cellular model with the data-assimilation method, we were able to incorporate model-generated data into the model to constrain wave dynamics in a 1-D case. By focusing on the state of discordant alternans, we gave data assimilation a more challenging problem to solve.

Breeding provides superior initial conditions for ensembles in data assimilation since the bred perturbations are only based on the dynamics of the FK model, and it also helps study error growth. During the breeding experiment, we did not find model times that had an high error growth in the FK model. We also found a relationship between the error growth between u and v in the breeding experiments. Error growth rates had a slight temporal lag between the two variables. We can use this relationship when real-world data begins being incorporated to the data assimilation process. We can use values of u, which we can observe, to help make corrections in v, which cannot be observed. The breeding experiments also showed low growth rates in the model for all breeding interval lengths, pointing to the FK model having no or little chaos. Because of this, only one time interval length for data assimilation was necessary.

Finally, results from data assimilation were presented. The main methods of evaluation of these results use the spread of the background and analysis ensembles and time series showing the difference in synthetic observations generated by the FK model and the resulting analysis estimates from data assimilation. We found the ensemble size that made the best balance between accuracy and computation time for this problem to be 20. This size is able to approximate wave dynamics in scenarios that 10 ensemble members failed to work. We found little difference in accuracy between using 20 and 100 ensemble members. For computation purposes, there is then little benefit for increasing the size beyond 20 ensemble members in 1-D, there may still be a need in 2- or 3-D. Using data assimilation with random initial background ensembles which will quickly lead to wave termination with just the FK model, were able to give better state estimates than a non-random initial background estimate free run with the FK model. This shows that data assimilation does a better job than the FK model does on its own. The multiplicative covariance inflation factor, ρ, was found to give the best results for correcting the spatial lag caused by ensemble collapse when $\rho = 2$. This method for dealing with

ensemble collapse is also computationally superior to additive inflation. Successful approximations of realistic observations were also made with good initial conditions, but not with random initial background ensembles. The parameter estimation application of data assimilation was done, however there was no success in recovering parameter values or in approximating wave dynamics.

Additional work will include using real-life observations as part of the forecast from data assimilation. These observations will have different error properties that are not possessed by model-generated data. Electrical cardiac models are only qualitative representations of true wave behavior. Some behaviors seen in experiments can be difficult to accurately represent. Therefore modeling assumptions and model detail will affect differences between data used in the simulations for this thesis and real-world experimental data. Other opportunities for future work lie mainly in extending this approach into 2- and 3-D. In doing so, results will extend beyond the cellular level. The higher dimensional cases will most likely require the ETKF algorithm used for the 1-D setting to be adapted to include the localization step. This is because domain sizes will be larger and an anticipated higher level of chaos in the larger dimensions.

References

[1] (2013). 2014 Heart and Stroke Statistical Update. *American Heart Association.*

[2] Chugh, S. S., & Reinier, K., et al. (2008). Epidemiology of sudden cardiac death: Clinical and research implications. *Progress in Cardiovascular Diseases*, 51(3), 213-228.

[3] Cardiac Arrhythmia Suppression Trial II Investigators, (1992) Effect of the antiarrhythmic agent moricizine on survival after myocardial infarction. *N. Engl.J. Med.* 327, 227-233 .

[4] Fenton, Flavio H., Elizabeth M. Cherry, et al. (2002). Multiple Mechanisms of spiral wave breakup in a model of cardiac electrical activity. *Chaos.* 12.3, 852-863.

[5] Fenton, Flavio, Elizabeth. M. Cherry, et al. (2002). Validation of realistic 3D computer models of ventricular arrhythmias with optical mapping experiments. *Pacing Clin. Electrophysiol.* 24, 538.

[6] I. Banville and R. Gray. Effect of action potential duration and conduction velocity restitution and their spatial dispersion on alternans and the stability of arrhythmias. unpublished.

[7] Watanabe MA, Fenton FH, Evans SJ, Hastings HM, Karma A. (2001) Mechanisms for discordant alternans. *Journal of Cardiovascular Electrophysiology.* 12, 196-206.

[8] Fenton, F., & Karma, A. (1997). Vortex dynamics in three dimensional continuous myocardium with fiber rotation: Filament instability and fibrillation. *Chaos*, 8, 20-47.

[9] Bueno-Orovio, Alfonso, Elizabeth M. Cherry, and Flavio H. Fenton. (2008). Minimal model for human ventricular action potentials in tissue. *Journal of Theoretical Biology.* 544-560.

[10] G. W. Beeler and H. Reuter. (1997). Reconstruction of the action potential of ventricular myocardial bres. *J. Physiol.* London. 268, 177210.

[11] Fenton FH, Cherry EM. (2008). Models of cardiac cell. *Scholarpedia* 3: 1868.

[12] Bullock TH, Orkand R, Grinnell A (1977). Introduction to Nervous Systems. New York: W. H. Freeman. ISBN 0-7167-0030-1.

[13] Jazwinski, A. H. (1970). Stochastic Processes and Filtering Theory. *Academic Press.*

[14] Toth, Zoltan, and Eugenia Kalnay. (1997). Ensemble Forecasting at NCEP and the Breeding Method. *American Meteorological Society.* 3297-3318.

[15] Patil, D.J., Istvan Szunyogh, et al. (2003). Local Low Dimensionality and Relation to Effects of Targeted Weather Observations. *University of Maryland.* 1-11.

[16] Szunyogh, Istvan, Eric J. Kostelich, et al. (2004). Assessing a local ensemble Kalman filter: Perfect model experiments with the NCEP global model. *University of Maryland.* 1-17.

[17] Ehrendorfer, M. (1994). The Liouville equation and its potential usefulness for the prediction of forecast skill. Part I: Theory. *Mon. Wea. Rev.,* 122, 703-713.

[18] Ehrendorfer M., and J. J. Tribbia. (1997). Optimal prediction of forecast error covariances through singular vectors. *J. Atmos. Sci.,* 54. 286-313.

[19] Kerr, Richard A. (2012). Weather Forecasts Slowly Clearing Up. *Science.* 338. 734-737.

[20] B. R. Hunt, E.K., Szunyogh. (2007). Efficient Data Assimilation for Spatiotemporal Chaos: A Local Ensemble Transform Kalman Filter. *Physica D* 230. 112-126.

[21] Hager, William W. (1989). Updating the inverse of a matrix. *SIAM Review* 31,2. 221239.

[22] Baker, L. (2007). Properties of the ensemble kalman filter. (Master's thesis).

[23] Whitaker, Jeffrey S., Thomas M. Hamill, Xue Wei, Yucheng Song, Zoltan Toth. (2008). Ensemble Data Assimilation with the NCEP Global Forecast System. *Mon. Wea. Rev.* 136. 463482.

[24] J. S.Whitaker, T.M. Hamill. (2002). Ensemble data assimilation without perturbed observations. *Mon. Wea. Rev.* 130. 19131924.

[25] Hoffman, Matthew J., Takemasa Miyoshi, Thomas W. N. Haine, Kayo Ide, Christopher W. Brown, Raghu Murtugudde. (2012) An Advanced Data Assimilation System for the Chesapeake Bay: Performance Evaluation. *J. Atmos. Oceanic Technol.* 29. 15421557.