THE DEVELOPMENT OF A DATA-DRIVEN MODEL CALIBRATION METHOD FOR PLASMA PHYSICS APPLICATIONS

A Thesis

by

CHRISTINE MARIE GREVE

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee,	Kentaro Hara
Committee Members,	Manoranjan Majji
	David Staack
Head of Department,	Rodney Bowersox

October 2019

Major Subject: Aerospace Engineering

ABSTRACT

The development and subsequent studies regarding statistical convergence for a data-driven calibration approach to physics-based models are presented with the ultimate goal of using the resulting method to analytically quantify anomalous behavior seen in experimental data of Hall effect thrusters. This calibration approach uses a single output signal to calibrate unknown input parameters of a computational model to a reference solution, either trusted analytical results or experimental data. The dimension of the output signal is increased by taking a time-delay embedding, based on the Takens Embedding Theorem, and the resulting time-lag phase portrait is binned as a probability distribution function. The first Wasserstein metric is used to quantify the difference between two solutions as a single variable. This process is automated using an evolutionary algorithm function from Sandia National Laboratory's DAKOTA algorithm. The canonical chaotic Lorenz attractor, a zero-dimensional bulk plasma model, and a two-dimensional Hall effect thruster model are used to characterize and minimize the numerical uncertainties incurred by this model calibration method and give conditions for the definition of an optimal solution. Results indicate verification of the method's ability to uncover unknown input parameter values. In particular, the model calibration method is shown to obtain results within 1% of the reference solution for various signals that were not used during the calibration process. Additionally, a more active, online, calibration technique is developed in conjunction with this thesis to detail the first step in the development of a more robust method in future work.

DEDICATION

To my family and friends, who constantly remind me that there is more than one way to land on the moon.

ACKNOWLEDGMENTS

It would take too long to thank every person who deserves to be thanked, but know that if we have come into contact in the last year, you deserve to be in this section.

First, I must thank my advisor Dr. Ken Hara. You are the reason I originally came to Texas A&M. Your confidence in me and what I have to offer as a researcher has been some of the best motivation over the past two years. Your patience as I figured out graduate school and computer modeling rivals that of some of the best since I can take quite a while to learn. You have always been there to encourage me and answer my questions no matter how many times it took to get the answer through my head. Thank you for letting me chase this rabbit that is data-driven modeling, and learning about this field side-by-side with me.

To my family, who have spent the past two years helping me tackle hurdle after hurdle, thank you. I know my habit of calling after a long day is not always the most exciting or talkative, but the support I have received is unmatched by anyone else. Even from thousands of miles away you found ways to make my day and help me jump through the next hoop. You are always the first in line to encourage me and remind me how much I have accomplished when I forget. You may never have understood me when I spoke about my research, but you always tried and that means more to me than I can express. Who would have guessed that the little girl who spent her free time writing short stories about space exploration would end up with a master's in Aerospace Engineering?

To my dearest labmates who have been with me the longest, Adnan and Rupali, you have been in the office every bit as much as I have. Your constant friendship and understanding on late nights and through long group meetings has been a blessing. I have so enjoyed hearing about your research and sharing mine with you. Thank you for making our windowless office a little brighter with your laughter and encouragement.

To my friends who have been there since the beginning of this journey, Jeppesen, Hunter, Tyler, Carl, and eventually Nooner and Laura. Your quirks and personalities have brought so much light and enjoyment to this experience that without it, I am unsure how I would have made it through

the stress. To everyone who has come since, you deserve every bit as much gratitude. Many of you have come from dancing and I thank you for reminding me that no matter what happens in school, no matter where I end up in life, I am still worth something. To my ballet students, old and young, thank you for showing me that teaching is not impossible and that I don't just have to pick one dream to chase.

Lastly, to the man who simultaneously took me under his wing and welcomed me as a colleague some five years ago. To the man who taught me more about life and electric propulsion than pretty much anyone else. To the man who to this day is one of my dearest friends. Kurt, I can honestly say that I do not think I would be where I am today without the impact you have had on my life, professionally and personally. Thank you.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a thesis committee consisting of Professor Ken Hara and Professor Manoranjan Majji of the Department of Aerospace Engineering and Professor David Staack of the Department of Mechanical Engineering.

The basic overview of the model calibration method was originally proposed to me by the Air Force Research Laboratory as a summer internship project in 2018. The Hall effect thruster model analyzed for Chapter V was provided by the Air Force Research Laboratory. The pre-created Galaxy Simulation Builder used for some of the initial testing was created by Stellar Sciences LLC. for the Air Force Research Laboratory.

The code utilized for comparison in Chapter VI was developed by Manoranjan Majji and Ken Hara and is used here with permission.

All other work conducted for the thesis was completed by the student independently.

Funding Sources

This graduate study was supported by a fellowship from Texas A&M University, an internship with the HPC modernization program and the HPC Internship Program sponsorship through the Air Force Research Laboratory, and a dissertation research fellowship from the NDSEG Foundation.

NOMENCLATURE

A, B, C	constants of the ionization rate coefficient equaiton
B	magnetic field
E	electric field
F_{ij}	a fluid-like flow of numerical mass
L_{ch}	thruster channel length
M_i	mass of a Xenon ioni
N_i, N_n	number density of ions and neutrals
N_{int}	initial number density of neutrals
N_s	number of time steps
$R_{\triangle}$	thruster channel width
T_e	electron temperature
U_i, U_n	ion and neutral speeds
$U_{i,w}$	ion acoustic speed
V_d	voltage
a_n	amplitude of the sinusoidal temperature function
d_{ij}	distance between bins of the same width on two separate distributions
e	elementary particle charge
f_{ij}	transportation cost
m, m_e	mass of an electron
t	time
$u_{e,\perp}$	electron bulk velocity

$\vec{x}_{\mathrm{ref}}$	reference solution data
$\vec{x}_{\mathrm{trial}}$	trial solution data
x	nondimensional rate of convection
y	nondimensional horizontal temperature variations
z	nondimensional vertical temperatrue ratio
Σ_X, Σ_Y	total weight of distribution X and Y
α	electron mobility parameter
β	nondimensional ratio of critical and current Rayleigh numbers
$\vec{\beta}$	set of unknown input parameters
ϵ	difference between two distributions
ϵ_0	convergence criterion
ϵ_{ion}	ionization rate coefficient
μ_{eff}	effective cross-field mobility
ρ	nondimensional geometric constant
σ	nondimensional Prandtl number
$\omega_{c,e}$	electron cyclotron frequency
ω_q	frequency of a sinusoidal temperature function

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

1. INTRODUCTION

The dynamics of spaceflight allow for electric power to be used to accelerate propellant for thrust in a manner more efficient than chemical-based propulsion. One method of generating thrust is to use the interaction of a current in an ionized gas and a magnetic field supplied by permanent magnets or electromagnetic coils to accelerate the ionized gas out of the thruster. Hall effect thrusters represent one branch of this electromagnetic propulsion. Developed since the 1960s, this particular type of propulsion is predominantly used for station-keeping or orbit adjustments on unmanned satellites, though some missions such as NASA's Dawn mission have used Hall effect thrusters as the primary source of propulsion [1]. These thrusters operate on the principle of accelerating ionized plasmas to high speeds and ejecting them out the open end of a thruster [2]. Figure 1.1 provides a visual guide of the basic Hall thruster operation. The back of the thruster houses the anode and the propellant gas injectors. The electrical counterpart of the anode, the cathode, is located either just outside the thruster channel [2], as seen in Fig. 1.1, or is centrally located within the channel ring [3] to provide a steady stream of electrons to the thruster. The electrons interact with an imposed radial magnetic field provided by coiled magnets within the thruster. The interaction of the radial magnetic field and axial electric field generates a Hall current where the electrons are trapped in a circling (azimuthal) ring in the downstream portion of the thruster channel. As the injected gas gets ionized, the Lorentz force, the cross product of the magnetic and electric fields, accelerates it through the thruster channel. Though the generated thrust is low as only ionized particles are ejected at high speeds, this type of thruster offers high specific impulse, throttleability, and high levels of efficiency [2].

The modeling of Hall effect thrusters has been studied for nearly as long as they have been developed. Models can follow individual particles [4], fluid continua [5], or a combination of the two [6] to develop the resulting plasma physics. The limitations of physical testing, from actual costs to the intrusion of probes or other data collecting methods in the thruster plume, have pushed computational models to advanced levels [7]. Despite the slew of complex models employed by

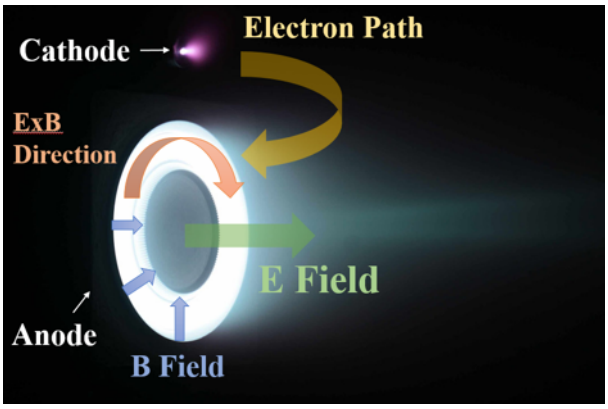

Figure 1.1: The basic operation principles of a Hall Effect Thruster.

the research community, the development of a truly predictive model is still in progress as most models have reached the same analytical limit where no further progress can be made without a better understanding of certain physical processes. One of these processes is the termed anomalous electron transport, where electrons are known to cross magnetic field lines and significantly alter the efficiency and lifespan of a thruster by bombarding and eroding plasma-facing surfaces [8]. Significant levels of electron mobility have been seen in experimental results that do not appear in any computational model to date, prompting strong focus on this research topic.

Part of the difficulty in developing a model for anomalous electron transport is the chaotic nature from which it is borne. The reliability of a computational model to capture physical behavior is dependent upon the model's ability to reproduce validated physics. The inherent complexity of physical phenomena proves difficult to reproduce in any field of research due to the close coupling of equations, existence of chaotic behavior (turbulence), and multiple temporal and spatial scales existent in the model. Specifically, modeling chaotic behavior is nearly impossible using only strict analytical expressions, as evident by the difficulty had by any research community in reliably reproducing chaotic occurrences in nature. To combat the complexity of natural phenomena, approximations are often made in computational models to simplify the analytical expressions being solved and reduce simulation time. These approximations introduce numerical uncertainties and can dampen significant physical processes. One method to counter these effects is to develop a

data-driven model that uses validated experimental or analytical results to inform relations between inputs and outputs of a computational model. In data-driven modeling, an arbitrarily complex map that is not necessarily derived from strict analytical expressions of physical behavior is developed to more accurately predict physical phenomena. These relations often operate as black boxes that are considered insufficient to understand fundamental physics due to the lack of strict analytics. By using data-driven modeling with physics-based analytical expressions that contain unknown parameters, also termed theory-guided machine learning, the accuracy of the model can be improved by restricting the learning process to only closure terms of the physics-based conservation laws. This technique has yet to be employed by the Hall thruster community but it can likely offer new insights into the physics if properly employed.

This thesis details the development, uncertainty quantification, and initial application of a static data-driven model calibration method before presenting a more active calibration method at the end of this work. The development of this calibration method, a steady-state or *static* method, is a critical first step towards a more robust and complex *dynamic* method that accounts for time-varying parameters. Developing a simpler system can help the community to understand the best optimization practices for the given system, whether the system can be optimized, the effects of noise on the system, and the difficulties in corroborating experimental and analytical data. After a literature review of Hall effect thruster models and data-driven modeling of dynamic systems, Chapter II will detail the actual calibration method used in this work. Chapter III discusses initial application to a canonical chaotic model, the Lorenz problem, to discuss some sources of numerical uncertainty. Chapter IV uses a zero-dimensional predator-prey model of a bulk plasma to perform various convergence studies on the proposed calibration method. Chapter V will see the method applied to one of the common Hall effect thruster models used today. Lastly, Chapter VI will present an active optimization process from the controls field that uses an extended Kalman filter to preform real-time estimation of simulation parameters.

1.1 Literature Review

The literature review will be split into two categories: (1.1.1) Hall effect thruster modeling and the specific behavior of anomalous electron transport and (1.1.2) data-driven modeling as applied to dynamic, time-dependent systems.

1.1.1 Hall Effect Thruster Models

Many Hall effect thruster models have been developed over the years to simulate different aspects of thruster behavior depending on the desired physics to be captured, the computational cost of the modeling technique, and the acceptable limit of numerical inaccuracies accrued by the chosen modeling technique. Kinetic particle-in-cell (PIC) models track macroparticles to study particle-level interactions [4, 9]. These models have been used by Campanell to study sheath instability [10], Katz for ion acoustic waves and anomalous electron transport [4], a three-dimensional model by Taccogna and Minelli [11] as well as one by Hirakawa that showed azimuthal oscillations' effect on electron motion [12], and a model by Fife that was used to compare PIC results with experimental data [13]. Fluid models treat each particle specie as a fluid continuum [5] and are used as a balance between simplicity and reliability in the model to show that the fluid-like response of plasma can be used to explain many phenomena seen in experiments [14]. Kaganovich et. al. use nonlinear fluid models to describe instabilities in Hall thrusters [15], Barral et. al. study electron-wall interactions and backscattering [16], Barral and Ahedo study the interaction of standing waves and the transport of neutral species [17], and Mikellides and Katz study the effects of the resistance to classical electron transport in the parallel direction [18]. Many fluid models are used as part of hybrid-PIC models that treat electrons as fluids and the other heavy species as particles [6]. Hofer et. al. use this type of model to develop a three-region electron mobility model [19]. Jorns proposes a transport model based on plasma wave theory [20], and Hara's fluid model has been used to study plasma perturbations [21]. Fife's hybrid model has been successfully used to identify breathing mode oscillations in plasma behavior [13].

Electron transport has specifically been studied by Hagelaar as a two-dimensional hybrid model [22],

Boeuf and Garrigues consider electron-neutral collisions dominant [23], Morozov et. al. and Latocha et al proffer that electron-wall interactions play a significant role [24, 25], Adam et. al. link plasma turbulence to anomalous electron transport [26]. Koo proposes a two-region spatial model of anomalous electron transport [27], and Hofer uses a three-region spatial model [19]. Fife's HPHall model has been edited to include an anomalous electron transport model based on Jorns [6, 20]. Note that this is not an exhaustive list of models used by the community, but a list of those pertinent to this work.

1.1.2 Data-Driven Modeling

Data-driven modeling is employed across a range of disciplines to further computational models by using trusted reference solutions to "train" a model [28, 29, 30, 31, 32]. Often a set of input parameter values are desired to be found such that they yield specific output characteristics. Various techniques that have been developed towards this end include symbolic techniques [33, 34], sparse regression [35, 36], compressive sensing [37], neural networks [38], system identification [39], and dynamic mode decomposition [40]. Disciplines including water reservoir research use data-driven modeling to provide insight to indicators or precursors of flow regime transitions and wave instabilities [31, 32]. Though machine learning and data-driven modeling are becoming common in various research communities, it is not very common in plasma physics. Most notably, Jorns has used symbolic regression techniques to create a steady-state data-driven model for anomalous electron transport [20]. More useful to this work is the plethora of research performed on nonlinear, chaotic systems and identifying underlying equations [41] or create symbolic equations from time series data [42]. Prediction of future chaotic attractor states by kernel regression is performed by Nadarya [43]. He, Wang, and Liu used particle swarm optimization to estimate parameters for chaotic systems [44]. Guerra et. al. performed nonlinear identification of a chaotic system using a radial basis neural network [45].

More widely studied is the use of data-driven modeling on dynamic, nonlinear, chaotic mathematical models. The history of phase space reconstruction, a method of viewing nonlinear dynamic data discussed later in this work, can be traced back to Poincare and a desire to model nonlinear

dynamics [46]. Kutz, Brunton, et. al. have used a variety of approaches for nonlinear systems including the Koopman operator [47], sparse identification [48], sparse regression [49], and dynamic mode decomposition [50] for nonlinear dynamical systems to reduce the amount of training data needed for calibration.

Though work continues to be performed towards a solution for anomalous electron transport, a majority of current approaches rely on a steady-state assumption where the mobility is based upon one or more constant-value parameters or weighting schemes. This may be an oversimplification of the steady-state operation mode of Hall effect thrusters. Instead, a dynamic (in time or space) solution must be determined to account for all modes of a Hall thruster that require different operating conditions. Work to date, such as that by Koo or Garrigues, has only assumed static models [19, 27]. The static model developed as part of this thesis serves as the building block for the development of a dynamic calibration model. The simple lack of data-driven modeling techniques applied specifically to plasma propulsion leave a broad field of potential new insights to computational models.

2. DEVELOPMENT OF A DATA-DRIVEN METHOD[1]

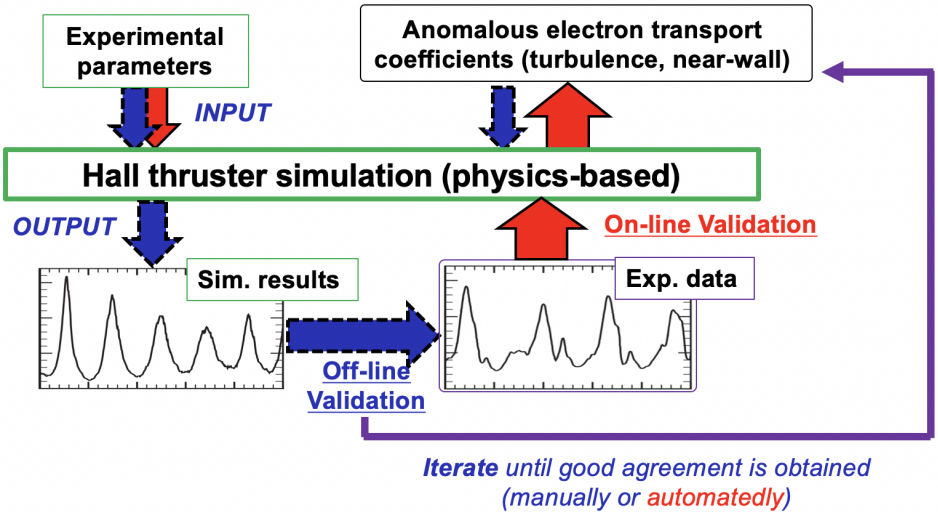

Figure 2.1: Overview of an online and offline data-driven model calibration method to depict the difference between methods.[51]

Data-driven modeling can take one of two forms: *offline* or *online*, shown in Fig. 2.1. Offline validation requires the full simulation to be run for all possible input parameters, iterating through simulations until good agreement in the observed output signal is obtained with the reference signal. Alternatively, online validation actively updates a random set of input parameters by comparing the output signal trace to the reference solution every few time steps within a single simulation to obtain a final signal trace of the desired quantities and, by extension, the corresponding unknown input parameters. Despite the difference in approach, both methods face similar challenges. One is the identification of relevant optimization metrics. It must be determined whether an optimized solution truly approaches a physically correct solution despite numerical inaccuracies incurred by

[1]Reproduced from C.M. Greve, K. Hara, R.S. Martin, D.Q. Eckhardt, and J.W. Koo. "A data-driven approach to model calibration for nonlinear dynamical systems" *Jour. of Appl. Phys.* Vol. 125, No. 244901, 2019. with permission of AIP Publishing.

the calibration method. It is important to note that true convergence may be impossible in all but an infinitesimal limit, especially for noisy and chaotic data, thus requiring a need to demonstrate statistical convergence. This comes from the concept of the Lax equivalence theorem [52] that determines whether results obtained from a numerical model converge to a unique exact solution in the limit of some infinitesimal discretization, spatial or temporal. Another challenge to both optimization forms is the selection of the output signal for calibration that best represents the system. A balance must be found between uniqueness of the signal chosen, a gradient in the response surface towards this solution, and the data analysis required. Additionally, a signal that affects the desired phenomenon in a system should be chosen to better predict appropriate input parameters or their time-dependent traces.

The resulting input parameters determined by a model calibration method are used to classify the type of optimization performed, and fit into two classifications: *static* and *dynamic*. Static optimization yields a stationary set of optimized parameters while dynamic optimization yields parameters that vary spatially, temporally, or both [54]. Initial static optimizations can be used to assess the validity of a given model or determine the best configuration of experimental data to be used before applying the more complex dynamic optimization method [55, 56]. For the case of electric propulsion, the concept of a steady-state operation mode has left some research relying heavily on static methods to only marginal success [19, 27]. Because of the variability of this device even during steady-state operation modes, dynamic time- and space-dependent equations and optimized solutions may be required to solve the anomalous behavior of Hall effect thrusters rather than the time-averaged solutions currently being pursued. As a first step towards the development of a dynamic optimization method, a static method is presented in this work.

In a general offline optimization scheme, the user's quantities of interest from either a known solution or observable datasets of the system are selected as the reference data, $\vec{x}_{\mathrm{ref}}$. While a data-driven model works using an arbitrary number of observable signals, one of the advantages is the possibility to construct a unique shadow manifold that directly corresponds to the underlying state-space dynamics of an entire system using only a selected subset of output signals [53]. Determining

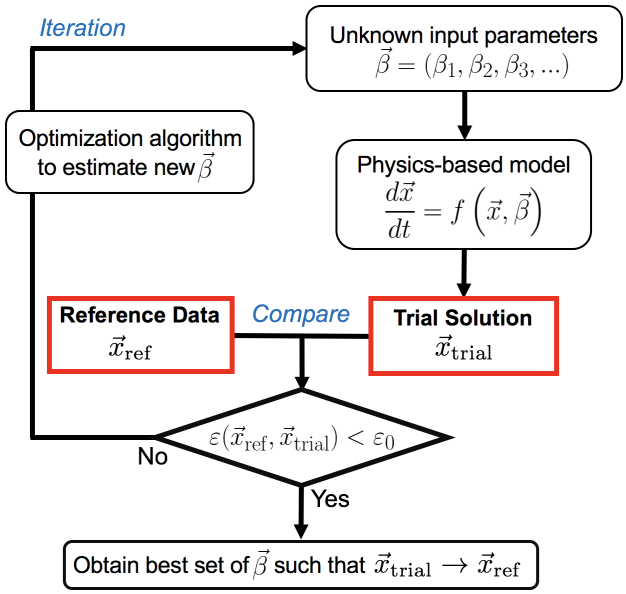

Figure 2.2: Overview of a data-driven model calibration to obtain appropriate input values that result in a near-reference solution. ε is the cost function, or the discrepancy, between the reference data, $\vec{x}_{\text{ref}}$, and the trial data, $\vec{x}_{\text{trial}}$ from a physical model of choice.

the necessary output signals directly translates to a basis for the reference data acquired from experiments. This would be beneficial to experimental work that often observes all possible state variables, a costly practice. Knowledge of the driving equations of a system can guide the selection of a subset of observable quantities that can capture the complete state space dynamics [57, 58], but determining the driving equations falls under the category of system identification which is not the focus of this thesis. The governing equations for the systems presented in this work are considered known and understood.

Figure 2.2 depicts the generalized data-driven optimization process used in this paper. The physics-based model is run with a selected set of values for the unknown input parameters, $\vec{\beta}$, to produce a trial solution, $\vec{x}_{\text{trial}}$. The trial solution is compared to the reference solution, $\vec{x}_{\text{ref}}$, and the discrepancy, ε, between $\vec{x}_{\text{trial}}$ and $\vec{x}_{\text{ref}}$ is calculated. If the discrepancy, called the cost function, is large, a new set of input parameters is determined by the overarching optimization scheme and the process repeats. This iterative cycle continues until a convergence criterion uniquely specified

for the given problem is met, i.e., $\varepsilon \leq \varepsilon_0$. When the minimum cost function is found, it can be considered that an (approximately) optimal set of input parameters which capture the dynamics of the reference solution is obtained. Determining a converged solution requires an appropriate distance measure between two solutions as well as a valuable convergence criterion. Additionally, an optimization algorithm that may not be able to rely on the existence of a coherent gradient in the response surface of the cost function is required to comprehensively search the domain of possible input parameter values for an optimized solution. The remainder of this section will develop the data-driven calibration method and address some concerns that arise from this general optimization process.

2.1 Representation of a Dynamic System

Dynamic, time-dependent signals have been commonly represented using linear techniques that employ a superposition of static basis functions to capture the dynamics of a system. The most common of these are the various types of Fast Fourier Transforms (FFTs) that decompose an *N*-point signal into *N* individual signals. These individual signals are transformed into individual frequency spectra that are synthesized into a singular continuous spectrum. Various extensions have been developed to address the shortcomings of the basic FFT. One limitation of the FFT is its assumption of a finite dataset that contains one full period of a periodic signal whose endpoints are interpreted as continuous. Any non-continuous end points result in discontinuities that generate high-frequency components in the FFT that do not exist in the original system and can corrupt the resulting FFT. Time-windowed FFTs only use data over a specific length of time, multiplying the data by a smoothly-varying function that decreases towards zero at the edges [59]. This reduces the amplitudes of the data at each end of the window, decreasing the size of the discontinuities. This can also be applied as a sliding window to capture all of the data as a series of windowed FFTs applied consecutively across a domain, though the length of the window limits the frequency resolution. Wavelet transforms decompose a signal into both real and Fourier space as compared to the FFT that only uses the Fourier space. This type of transform not only provides the spectrum of frequencies present in a system, like an FFT, but also gives information as to when the changes

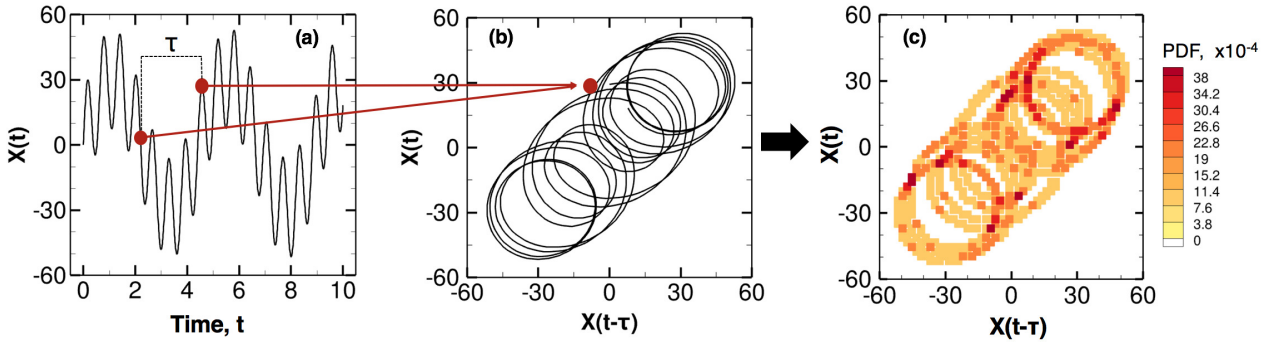

Figure 2.3: (a) A time-dependent solution. Shown is an example signal $X(t) = \sin(0.3t) + 30\sin(1.4t) + 23\sin(10t)$. (b) The solution converted into trajectories, the time-lagged phase portrait (TLPP), using the time lag, τ. (c) The probability distribution function (PDF) of the TLPP is obtained by binning the trajectories, which is referred to as the PDF-TLPP in this paper. 50 bins are used in each direction of the PDF-TLPP.

in frequencies occur.

Linear techniques can capture the spatio-temporal evolution of frequency components and wave modes of systems. However, these methods are subject to challenges when noise is present in the system, when the system needs to be represented succinctly, and when determining causality and correlation in a system [60]. Though mathematically the FFT can produce an exact reconstruction of the original signal, in practice, measurements are discrete and will contain noise. The fidelity of the representation is linked to the number of degrees of freedom in the FFT. Experimental noise typically inhabits higher frequency regions, which can only be captured as the number of degrees of freedom increase. Thus, FFTs either require very long signals or a truncation of the total number of modes generated to handle noise in a a system. The transforms provide full and often complex representations of dynamic systems that are difficult to simplify without losing the physics of the system. Linear techniques also lose the correlation between near points or frequency modes, preventing one's ability to understand causality in nonlinear systems. Conversely, nonlinear techniques can be used to represent a signal as a unitary symbol in n-dimensional space without immensely long signals in such a way that the data remains correlated and physical.

Figure 2.3 depicts an example of using a time-lagged embedding to create a time-lagged

phase portrait (TLPP). Figure 2.3 (a) is mapped against itself at an earlier time to create a two-dimensional time-dependent evolution of the system, shown in Fig. 2.3(b). The embedding separates unitary symbols from signal measurements by one or more fixed time delays to capture the state of the entire system in a single signal trace, i.e., an embedded phase portrait can be constructed using a signal at time (t) and at a delayed time $(t - \tau)$ in a coupled dynamic system, where τ is the specified delay. Each symbol in the phase space of the TLPP represents a complete state of the coupled system at that instant in time [60]. Dynamical systems theory consists of casually linked time-dependent variables of the same system. The metric, dynamical, and topological properties of the created strange attractor are characteristic of the underlying nonlinear dynamical system [61]. In the case of deterministic nonlinear dynamical (chaotic) systems, the variables share a common attractor manifold such that each variable can identify the state of the others because of its deterministic nature [60]. This implies that we can model the underlying dynamics of a system by modeling the dynamics of the corresponding points in the phase space, the foundation of predictive modeling. Using this principle, a discussion of the necessary delay embedding must be considered.

The conditions for which a single signal can be embedded to higher dimensions to obtain a unique map of the original discrete-time dynamic system are provided in Takens Embedding Theorem [62]. Takens proved that instead of the $2n + 1$ individual signals required by the Whitney theorem for uniqueness in phase space, the time-delayed versions of a single signal could suffice to embed a n-dimensional manifold. The theorem reads:

Theorem 1. *Let M be a compact manifold of dimension m. For pairs (ϕ,y), where ϕ:$M \to M$ is a smooth diffeomorphism (an invertible function that maps one differentiable manifold to another such that both the function and its inverse are smooth) and y:$M \to \mathbb{R}$ a smooth function, it is a generic property that the $(2m + 1)$ –delay observation map $\Phi_{(\phi,y)}$:$M \to \mathbb{R}^{2m+1}$ given by $\Phi_{(\phi,y)}(x) = (y(x), y * \phi(x), ...y * \phi(x))$ is an embedding; by 'smooth' we mean at least C^2. [62].*

Put colloquially, a smooth diffeomorphism between two manifolds and a smooth function can use $2m + 1$ delay embeddings to create a unique representation of system dynamics. Because the number of attractor dimensions is unknown for real systems and can be large since the amount of data required to densely populate the attractor grows with the power of its dimension, the desire exists to use fewer than $2m + 1$ embeddings.

In this work, only one time embedding is considered to construct a two-dimensional time-lagged phase portrait. It is understood that an optimal solution may not be uniquely defined in this low-dimensional TLPP even in the limit of infinite data. This could result in more than one set of input parameters yielding an optimal solution and require a multi-delay embedding to create a truly unique solution. Nevertheless, even a low-dimensional mapping can provide useful information for the optimization process and the underlying dynamics of a system.

In conjunction with determining the number of embedding dimensions to be used, a discussion of the selection of an appropriate time lag is required. From theory, as the amount of data approaches infinity for a finite dimensional attractor, the time lag used for the embedding becomes irrelevant so long as the density of points on the finite dimensional attractor becomes high faster than the lag approaches zero. However, in finite real-world data, the short time lags from measurement are those most influenced by experimental noise, while arbitrarily long time lags can corrupt the predictive power of the embedding because chaotic effects obfuscate casual relationships in data. At long lags, the subsequent points appear only stochastically related rather than the result of causal dynamics. A common approach to compute an optimal time lag τ is based on the amount of mutual information between a pair of observed values. The optimal time lag τ returns the mutual information between observations is lowest. For timescales shorter than the Lyapunov exponent, high mutual information is expected because of the strong deterministic dependence of the signal on its lag. However, at very long lags, chaotic systems become unpredictable and the mutual information approaches zero. The heuristic chosen by Fraser, the first minimum of mutual information, provides a generally good choice of lag for time delay embeddings because it distinguishes a point

where correlations resulting from short sampling time redundance are overcome by the mutual information contribution from relevant characteristic dynamic timescales of the system. However, this tradeoff is a delicate balance and the existence of such a minima is not guaranteed. Further discussions to this end can be found in Reference [64]. For this work, the time lag is chosen dependent upon the uniqueness of the resulting TLPP representation.

2.2 Binning the TLPP

The time-lagged phase portrait is still a direct portrait of the original system where each point is discrete and the number of data points is of the same order of magnitude as the original output signal. Validation of chaotic systems is challenging due to the randomness of state variable patterns in spatio-temporal evolutions. Yet, this randomness reduces the requirement to exactly retrace the exact system trajectories to obtain an optimal solution. This can actually help to reduce effects of noise in a system as the stochastic representation of the time a system spends in a discretized region of space becomes more significant than a precise spatio-temporal evolution. Binning the TLPP as a probability distribution function (PDF) provides a way to characterize nonlinear dynamic systems without exactly tracing the chaotic variations of state variables, enabling robust optimization of the system, Fig 2.3 (c). This binning greatly reduces the amount of data used in the calibration process as the amount of data is only as large as the number of bins used to discretize the continuous TLPP. Note that infinite data with an infinitesimally small bin size would yield a continuous PDF-TLPP of the time-averaged properties of a dynamic system.

For any finite data series, a probability density estimate will contain both sampling and discretization errors that will allow for only statistical convergence between solutions. The convergence of the low-dimensional PDF-TLPP is not sufficient to guarantee uniqueness of the solution but is a necessary prerequisite for convergence of higher-dimension systems. This makes the low-dimension PDF-TLPP useful for model calibrations, verification, and validation of data-rich, time-variant, nonlinear, chaotic systems.

Despite these sources of error, the PDF-TLPP yields three key benefits compared to more traditional linear methods. (i) Because the system does not rely on the signal being periodic or

continuous, there are no restrictions on using a non-smooth signal. Near-discontinuities are easily represented, which is vast improvement over gradient-based or spectral methods. (ii) Based on Takens Theorem, the proposed representation remains mathematically connected to the dynamics of the entire system for any number of output signals used. This is imperative to reduce the amount of data required for model calibration without compromising the uniqueness of the solution. (iii) Having reduced the data to a probability distribution function, a variety of classical distance-measuring techniques can be employed to quantify the difference between two solutions. The selected measurement for this work is discussed below.

2.2.1 Distance Measurement

Traditional optimization techniques work to minimize a specific value, typically a scalar mathematical norm for the case of model calibration. This norm must meet two basic criteria to ensure that the value, the objective or cost function, is properly minimized. The first condition is that two identical solutions should return a zero value. The second is that the greater the difference between two solutions, the larger the value of the norm should be. Distance measures are commonly used in the image-processing community to compare two images, and have been well researched and documented [65]. The most common mathematical norm used across a variety of disciplines is the L_2 norm, also known as the Euclidean norm. The L_2 norm is calculated as the square root of the sum of the squared vector magnitudes of distance, specifically, the shortest distance to move from one position to another:

$$||x||_2 = \sqrt{|x_1|^2 + ... + |x_n|^2}, \tag{2.1}$$

Because this method involves squaring the error values, outlier points hold larger weight and can be the downfall of this method as it tries to reconcile an outlier at the expense of other well-matched data. This becomes problematic when two solutions occupy entirely different regions of space and the resulting L_2 norm is deceptively low compared to two solutions that are far similar [66, 67].

This work uses the first Wasserstein metric, often employed by the image processing com-

munity as the Earth Mover's Distance [68] and the mathematics community as the Kantorovich-Rubinstein distance [69]. A similar process has been performed by Robim, Yiou, and Naveau [70] using the Wasserstein metric, W_1 metric. Simply stated, in the limit of a single delta function describing each of two distributions, the W_1 metric is the difference between the distributions defined by a cost metric derived from the Euclidean distance between two points in phase space [68].

Consider F to be a feasible fluid-like "flow" that represents the total weight of a distribution, e.g., the entire PDF-TLPP, and the distance said weight moves from one distribution location (X) to the other (Y). The total work performed by a flow can be written as,

$$w(F, X, Y) = \sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij} d_{ij}, \tag{2.2}$$

where f_{ij} is the transportation cost, i.e. the weight that is moved between the discrete points x_i and y_j in distributions X and Y, respectively, $d_{ij} = |x_i - y_j|$ is the distance between x_i and y_j, and m and n are the number of data points in each distribution. For this work, the cost f_{ij} can be considered the weight of a single discrete bin in one PDF-TLPP, e.g., the trial solution, that is moved some distance, measured as a Euclidean distance, to match a bin of similar weight in the other PDF-TLPP, e.g., the reference solution [71].

The Wasserstein metric is the minimum net work of Eq. 2.2, normalized by the weight of the lighter distribution,

$$W_1(X, Y) = \frac{min_{F=(f_{ij})} w(F, X, Y)}{min(\Sigma_X, \Sigma_Y)}, \tag{2.3}$$

where Σ_X is the total weight in distribution X and Σ_Y is the total weight in distribution Y. In order to remove the effect of the number of sampling points (e.g., time steps), the weights are normalized. The Wasserstein metric allows for complex situations such as partial mappings and comparing bins of different weights, as described in the work of Bonneel, *et. al.* [72].

Having reduced the difference between two solutions to a single value, the Wasserstein metric, an optimization algorithm can be employed to minimize this cost function. The existence of an

apparent gradient in the W_1 metric can assist in locating an optimal solution more rapidly. It can be seen from Eqs. 2.2 and 2.3 that a zero-valued W_1 will be returned if the two images are identical since $d_{ij} = 0$. It will be shown later that in the near-reference solution region, any gradient towards a true solution that exists breaks down and becomes too chaotic for the optimizer reliably use to determine where $W_1 = 0$. The convergence criterion, ε_0, should be low enough to allow for optimization but not so low that the system runs for thousands of iterations. The focus of this work is to minimize the cost function rather than meet a convergence criteria, thus ε_0 does not play a significant role in the remaining sections and is left for future work.

2.2.2 Optimization Algorithm

Having detailed a method to succinctly compare the output of a system to those of a reference solution and the difference reduced to a single value to be used with a classical optimization algorithm. The desire to automate the optimization process and run it on a supercomputer lead to the decision to use the Design Analysis Kit for Optimization and Terascale Applications (DAKOTA) optimization algorithm library developed by Sandia National Laboratories [73]. DAKOTA provides robust software for optimization algorithms and uncertainty quantification. This limited the possible methods to be used to those available in DAKOTA.

There are a variety of optimization techniques that can be used to obtain an optimal solution. Function-based methods such as interpolation or pattern searches rely on the specific algorithm to select points to be tested as part of a pattern, regardless of the objective functions. Gradient-based methods such as gradient descent or coordinate descent rely on the existence of a gradient in the response surface of the objective function towards a minimal solution. Hession methods such as Newton's method or interior points use the second-order derivative of the function to minimize the objective function. Nonlinear systems can exhibit mode transitions, yielding non-smooth and non-convex response surfaces during optimization. This precludes the use of gradient-based methods for more robust, though expensive, global optimization techniques. The development of a derivative-free global optimization method exceeds the scope of this work and complexity of the problems studied, thus a pattern search provided by DAKOTA is used: the evolutionary algorithm.

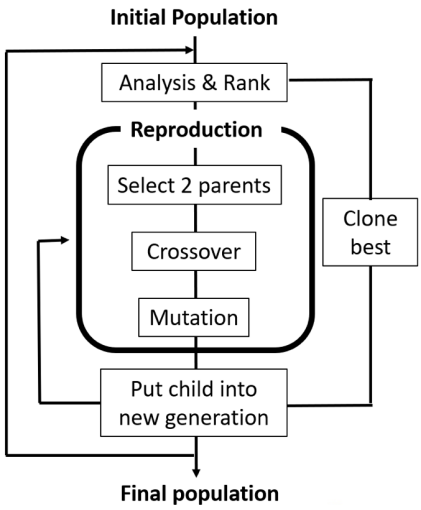

Figure 2.4: The flowchart for an evolutionary algorithm optimization scheme [74].

Figure 2.4 shows an example of the evolutionary algorithm that is selected for optimization, named $coliny_{EA}$ from the SCOLIB library in DAKOTA[73]. The initial population refers to the first generation of points that exist randomly throughout the domain space. Each point (member) of the initial population is ranked according to the value of each objective function, the W_1 metric values. The members with the lowest objective function values are cloned while the entire population is simultaneously split into pairs of points (parents). These pairs generate “children" points according to user-defined percentages specifying how likely points are to be similar (crossover) or to differ (mutation) from the parents. The new points represent a new generation (population) and the process continues until the user-defined convergence criterion is met by one of the trial points. Plotting the parameter sets tested as a function of time and as a function of their resulting W_1 metric values are instrumental to determine the fittest ranges for each input parameter for subsequent optimization attempts if desired. The evolutionary algorithm offers high concurrency, randomly searches across the entire domain initially, and results in a narrowed range of possible solution values.

A few details about the simulation process are included for completion. Initial work was performed using the Galaxy Simulation Builder, a graphic user interface for using DAKOTA devel-

oped by Stellar Science LLC and the Air Force Research Laboratory [74]. The interface allows the development of an automated optimization program for any arbitrary model without altering the source code as well as acting as the coordinating server for remote computing. It will be noted when simulations using the GSB are performed for certain optimization attempts. Whether the GSB is used or not, the DAKOTA algorithm is selected and customized with parameters including the maximum number of function evaluations, convergence criteria, input parameter search ranges, and how many evaluations can run concurrently. The code sections that run the model, generate the PDF-TLPP, and calculate the W_1 metric are written to accept the previous step's output as a necessary input file. The initial model can be of any coding language, this work uses C and Python depending on the system, while the modules corresponding to the creation of the PDF-LTPP are written in Python. Calculations of the W_1 metric require the PDF-TLPP to be normalized and the code to use linear programming techniques as this is inherently an optimal transport problem. The Python Optimal Transport (pot) library is implemented with the pre-developed emd function based on the work of Bonneel, et al. [72, 75].

3. INITIAL APPLICATIONS USING THE LORENZ PROBLEM[1]

Initial tests of the proposed calibration model are performed using the canonical nonlinear, chaotic Lorenz system. This dimensionless chaotic attractor, named for the stationary nodes around which the system orbits, was developed by Edward Lorenz to model atmospheric convection [76]. Lorenz developed these equations to emulate rolling fluid convection as an extremely simple model to visualize patterns noticed in weather that represented some strange order despite being chaotic. For certain constants, the system is bounded and non-periodic. The system will not diverge to infinity but will also not repeat any individual trajectories. Thus, while this is a purely deterministic system, it is susceptible to infinitesimal differences in initial conditions such that its location at a sufficiently long time is unpredictable. The non-dimensional three-equation system is given by

$$\frac{dx}{dt} = \sigma(y - x), \tag{3.1a}$$

$$\frac{dy}{dt} = x(\rho - z) - y, \tag{3.1b}$$

$$\frac{dz}{dt} = xy - \beta z, \tag{3.1c}$$

where x is proportional to the rate of convection, y to the horizontal temperature variation, and z to the vertical temperature ratio. The constant σ is the Prandtl number, the ratio of kinematic viscosity and thermal diffusivity, β is the ratio of the current and critical Rayleigh numbers, and ρ is a geometric scaling factor. Lorenz discovered that specific periodic solutions grew for Rayleigh numbers larger than a certain critical value and produced chaos. The instability of initial conditions is one of the earliest discoveries of the well-known butterfly effect in chaos.

The starting conditions chosen by Lorenz are still widely used in the literature today as they lead to the recognizable butterfly attractor shape. Constants of $\sigma = 10$, $\beta = 2.667$, and $\rho = 28$

[1]Reproduced from C.M. Greve, K. Hara, R.S. Martin, D.Q. Eckhardt, and J.W. Koo. "A data-driven approach to model calibration for nonlinear dynamical systems" *Jour. of Appl. Phys.* Vol. 125, No. 244901, 2019. with permission of AIP Publishing.

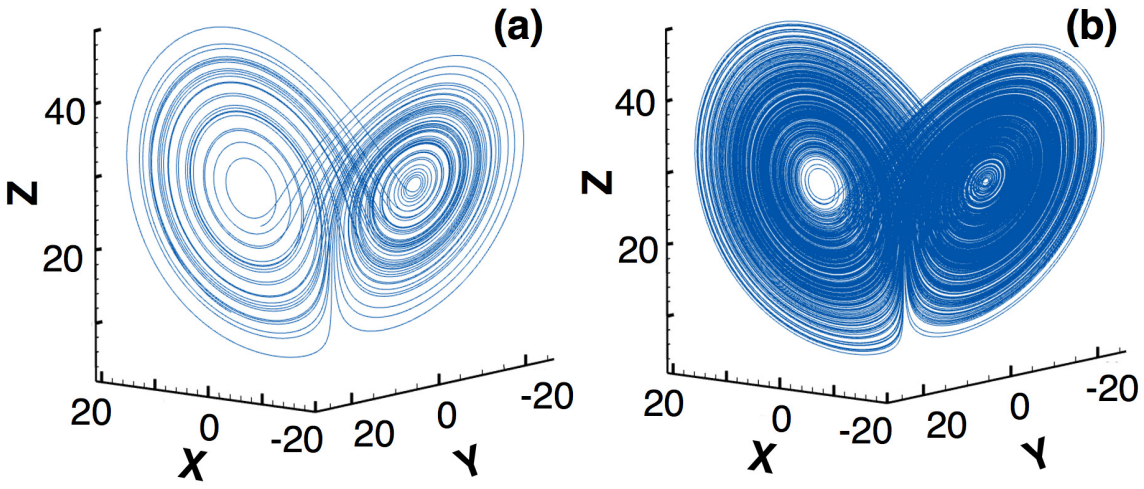

Figure 3.1: The Lorenz attractor for $(\sigma, \beta, \rho) = (10, 2.667, 28)$ and an initial condition of $(x_0, y_0, z_0) = (10, 10, 10)$ for different timesteps. (a) $N_s = 10^4$ and (b) $N_s = 10^5$ time steps while a constant $\Delta t = 5 \times 10^{-3}$ is used to integrate in time.

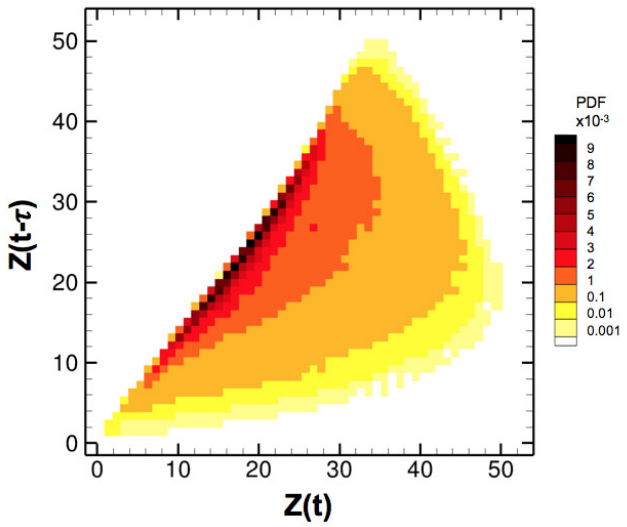

Figure 3.2: The Lorenz attractor PDF-TLPP as viewed from the z-direction.

and an initial starting point of $(10, 10, 10)$ are used as the reference solution for the following studies unless otherwise noted. Figure 3.1 shows the resulting attractor for (a) 10^4 and (b) 10^5 time steps based on a forward Euler integration method using a dimensionless time step of 5×10^{-3}.

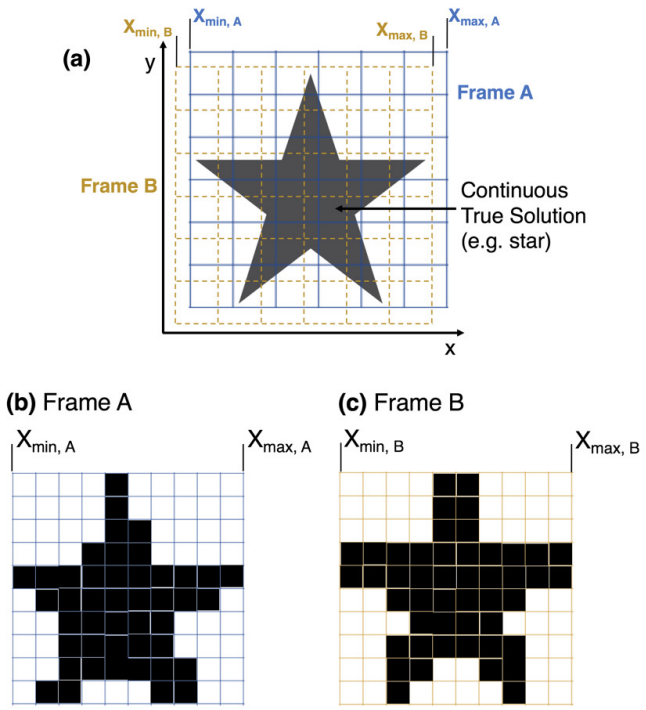

Figure 3.3: A depiction of how the choice of a frame viewing the same continuous true solution image can yield differently discretized PDF-TLPPs, resulting in a non-zero W_1 metric between images. (a) A depiction of the continuous true solution and the location of the two discretized frames. (b) The resulting discretized PDF for Frame A. (c) The resulting discretized PDF for Frame B.

The results show that the TLPP is essentially bounded and the trajectories are non-periodic, i.e. new trajectories appear in the longer simulation but the bounds stay relatively constant. Note that the attractor point locations remain identical, but the eye regions shrinks as the simulation is run longer. The resulting butterfly attractor can be transformed into a PDF-TLPP using the x, y, or z coordinates. For the sake of optimization, it is unwise to use the z-direction data. The attractor viewed from this perspective appears as a single lobe, seen in Fig. 3.2, which creates a non-unique image for optimization. Thus, the x-direction is chosen for all future studies.

3.1 Frame Dependence

The quality of the PDF-TLPP plays a critical role for the calculation of the W_1 metric due to numerical uncertainties brought about by changing a continuous solution to a discretized one. In

the limit of infinite data, both the bin size and variance per bin can theoretically be reduced to zero. However, finite data poses additional challenges in using the PDF-TLPP. To illustrate the numerical errors associated with the finite bin size of PDF-TLPP(s), consider a continuous true solution plot, e.g., a star, as shown in Fig. 3.3. A discretized PDF is constructed by binning the continuous solution based on a given area, referred to as a *frame* in this work, surrounding the continuous solution [77]. Two discretized frames, A and B, are considered in Fig. 3.3(a). The smallest and largest frame values, i.e., the bounds, are notated as x_{min} and x_{max} for each frame. The two frames are scaled to the same size and possess the same number of bins, but are offset in location. In other words, x_{max} and x_{min} differ but $x_{max} - x_{min}$ remains constant. The discretized (unnormalized) PDF is constructed in a binary fashion in this example so that a continuous solution in the discrete bins yields a value of 1 to be assigned to the bin; otherwise the value remains zero.

Figures 3.3(b) and 3.3(c) show the discretized PDF of the continuous solution for Frame A and Frame B, respectively. Since the bins have a finite width, the discrete locations used to contain the continuous solution vary between frames. Although the reconstruction of the discretized PDF is based on an identical continuous solution, it can be seen that the discretized PDFs are different, which yields a non-zero W_1 metric. If the width of the bins were reduced to zero and the number of data points was infinite, a continuous image would be created and the PDF-TLPPs would result in identical images despite their difference in frame location, yielding a zero W_1 metric. Hence, the quality of the frame properties introduce numerical bias errors to the optimization process. In a similar manner, changing the difference in frame such that $x_{max} - x_{min}$ differs will similarly affect the W_1 metric and introduce other numerical errors to the process.

For optimization schemes with large ranges of possible parameter values, the envelope or location of the signal can drastically change. If a *static*, non-changing, frame is used, the user runs the risk of obtaining misleading W_1 values due to part of the original signal falling outside of the set frame size. Using a *dynamic*, changing, frame allows the optimizer to view the full plot of the signal every iteration to ensure a true understanding of the response surface far from the reference solution. A possible practice could be that once the range of allowable values for each input pa-

rameter has been sufficiently reduced using a dynamic frame, the user can then implement a static frame to better compare overall structure of the different solutions. If a given trial produces an exact replica of the reference data except for a single outlying trajectory in the phase space such that the frame is affected, a dynamic frame PDF-TLPP would result in a poor W_1 value. The outlying trajectory alters the size (and bounds) of the frame and shift the location of the bins which affects the level of detail captured by the discretized image. Since the static frame employs identical bins (location, size, and bounds) for different images, the same level of detail can be retained for the main portion of the plot and the effect of the outlying trajectory can be significantly reduced.

3.2 Effects of Sampling Time and Selection of Frames for the PDF-TLPP

If the system is run infinitely long, the solutions would theoretically converge to the same image regardless of initial condition, meaning there would be no numerical errors associated with the finite data. Since an infinitely long calculation and data processing is not feasible, the effects of a discrete and finite length sampling time on the W_1 metric calculations must be understood. To quantify how the sampling time affects the resulting W_1 metric, a comparison study is performed where different finite numbers of time steps, N_s, ranging from 10^6 to 10^7 are compared to a reference solution, the result from 10^7 time steps. The test cases are exactly identical in their parameters (σ, β, ρ) and initial conditions (x_0, y_0, z_0), only differing in the number of time steps for each simulation. The time lag τ is set at 20 time steps while the discretized PDF-TLPP is binned using a 50×50 grid. The x-signal serves as the embedded signal for every run.

Figure 3.4 shows how the discretized PDF-TLPP changes as the sampling length increases using *dynamic* TLPP frames, in which the maximum and minimum bounds of the frame vary for each run while the number of bins is kept constant. A few noticeable differences can be seen in Figs. 3.4(a) and 3.4(b). (i) The maximum and minimum bounds are increased from ±21.7 to ±21.8 (see the axis values), while the number of discrete bins is kept constant. Hence, the numerical error caused by different frame sizes is introduced, as discussed in Sec. 3.1. (ii) A gap near the corresponding attractor point in the lower half region disappears with a longer sampling time since there are more trajectories in that area, as is shown in Fig. 3.1(b). Figure 3.4(c) shows

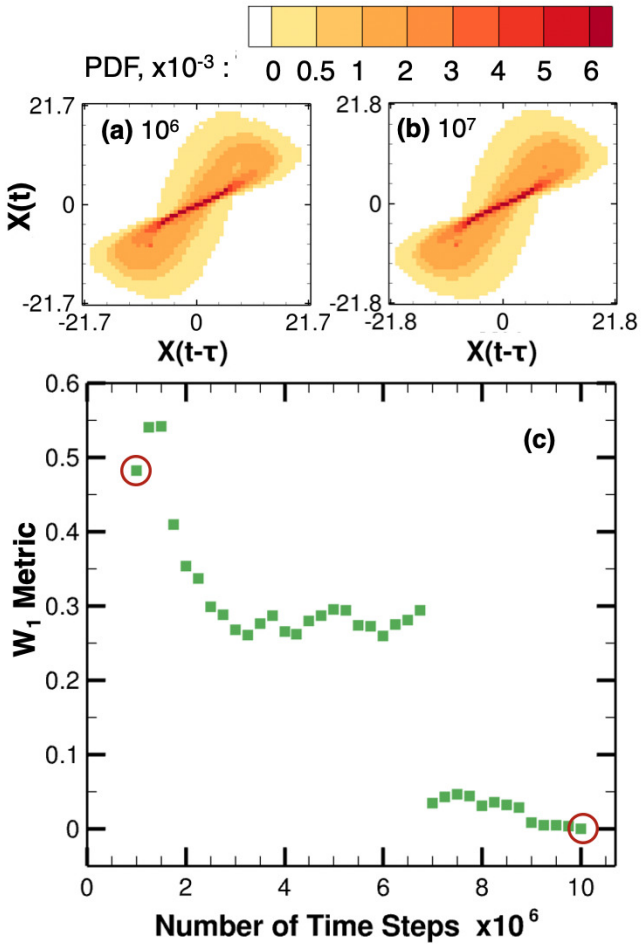

Figure 3.4: The effects of sampling time using the original *dynamic* frame for the baseline case. (a) the normalized PDF-TLPP for 10^6 time step run, (b) the normalized PDF-TLPP for 10^7 time step run, (c) W_1 metric values for a reference solution of 10^7 time steps using a dynamic frame. Note that the bounds of the PDF-TLPP in (a) and (b) are different, which results in a large W_1 metric, i.e., numerical error. The red circles in (c) denote the cases demonstrated in (a) and (b).

that the W_1 metric is large for a smaller sampling time, but approaches zero as the sampling time is increased. Note that the figure depicts a large discontinuous drop in the W_1 metric value from 0.3 to approximately 0.05 at $N_s = 7 \times 10^6$. This sudden decrease in W_1 metric corresponds to the "locking" of the bounds of the frame to the same size as the reference solution. When the frame size is unchanged, the only difference between PDF-TLPPs is the normalized value in each of the bins due to the number of trajectories. As the simulation is run longer, the normalization becomes

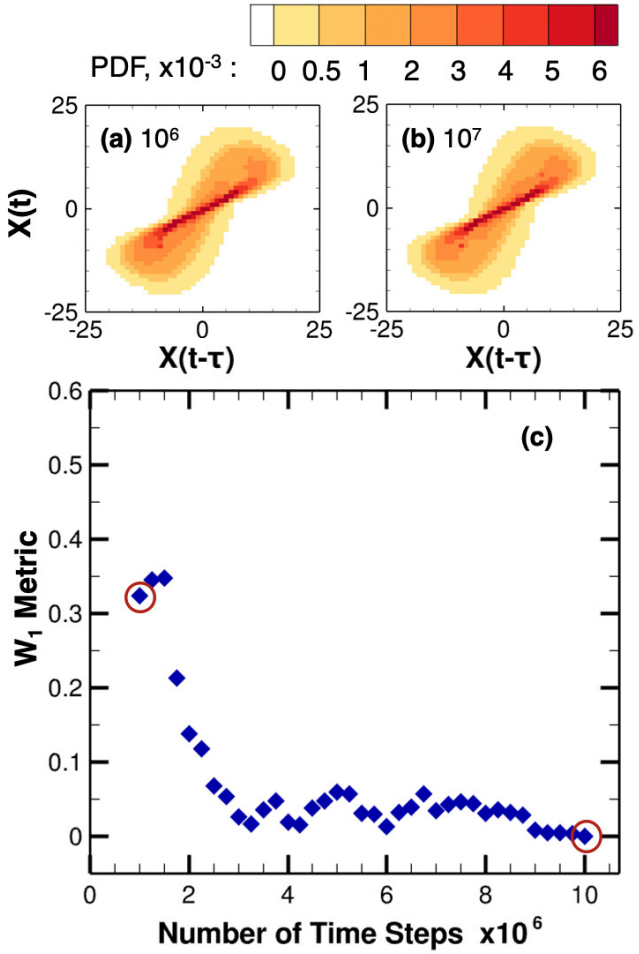

Figure 3.5: The effects of sampling time using the *static* frame for the baseline case, similar to the dynamic cases shown in Fig. 3.4. Note that the frame is fixed at a $X(t) \in [-25, 25]$ range for the static frame. (a) the normalized PDF-TLPP for $N_s = 10^6$ time step run, (b) the normalized PDF-TLPP for $N_s = 10^7$ time step run, and (c) W_1 metric values for a reference solution of $N_s = 10^7$ time steps using a static frame.

closer to the reference solution until the optimal solution is reached. This study gives evidence for the potential failure of the dynamic frame for an otherwise identical system (see Sec. 3.1: Frame Dependence).

Figure 3.5 shows the convergence study of N_s using *static* PDF-TLPP frames, where the maximum and minimum bounds are kept identical for all runs. While a larger frame may smear some of the finer structures in the TLPP when the same number of discrete frame bins is used for each

plot, it has been confirmed that the W_1 metric values are not significantly altered even when doubling the frame sizes, i.e. the maximum and minimum bounds. Figures 3.5(a) and 3.5(b) show the PDF-TLPP obtained for $N_s = 10^6$ and $N_s = 10^7$, respectively, when using a static frame of $X(t) \in [-25, 25]$. The original time-dependent data, $x(t)$, are identical between Fig. 3.4 and 3.5. The disappearance of the gap surrounding the attractor points seen in Fig. 3.4(a) but not in Fig. 3.5(a) is potentially the most noticeable smearing effect due to the use of different frames. As shown in Fig. 3.5(c), the W_1 metric values are lower than those of the dynamic frame, since there is no numerical error due to the frame misalignment. The set frame size also prevents a sudden drop in W_1 metric value observed in the dynamic frame results for the same reason. Instead, a relatively continuous decrease in the W_1 metric value can be observed as the sampling time increases, demonstrating the smooth convergence of the Lorenz's solution. Although the overall trend is decreasing, the decrease is not monotonic in nature, which is likely due to a low-frequency physical oscillation between attractors of the Lorenz system.

The static frame has distinct advantages over the dynamic frame. The static frame prevents the shifting discretization of bins from altering the discretization of the image. Also, the image is always viewed with the same resolution. The disadvantage of the static frame during optimization is that a wider frame must be selected to allow for any shifts in the actual trajectories. The Python model used in this work fails if any point in the generated plot falls outside of the set static frame, causing the DAKOTA optimization algorithm to terminate. For this reason, the static frame is only used once the allowable ranges of parameter values is significantly reduced to near-reference solution input parameters so the selected frame is not far from the true solution, avoiding the loss of detail in the system. Subsequent studies regarding the sensitivity of the W_1 metric to the range of axis chosen were performed with minimal difference in W_1 metric values for different frame bound ranges.

It is important to note here that theoretically, the W_1 metric should converge to zero as $t \to \infty$ because the weighting of the outliers becomes $1/\infty \to 0$, resulting in a transportation cost of 0 in the W_1 metric. However, for finite data, the trajectories alter the weighting of the discrete bins

and yield a non-zero W_1 metric between identical time-varying signals when different sampling times are chosen. Additionally, numerical models inherently face numerical inaccuracies such as discretization and round-off errors. In Lorenz systems, which are low-dimensional (in terms of degrees of freedom), other methods exist to perform model calibration with far less data than presented in this work [78]. Representing the entire system with minimal, finite data may reduce noise discrepancies when comparing two solution datasets due to the higher signal-to-noise ratio of a lower-dimension PDF-TLPP. Because of the desire to use this optimization method with a higher-dimension system, methods using less data are not studied at this time.

3.3 Effects of the Initial Condition

To view the trend of the W_1 metric with a physically different solution than the reference case, the initial condition is changed to $(x_0, y_0, z_0) = (10.01, 10.01, 10.01)$ for both the dynamic and static frames. This is called the non-ideal case as opposed to the $(x_0, y_0, z_0) = (10, 10, 10)$ ideal case. The reference solution is the data obtained for $N_s = 10^7$ time steps. Both the dynamic and static frame methods from Figs. 3.4 and 3.5 are used to show the difference in trend for a non-ideal solution.

Figure 3.6 shows the trend in the W_1 metric as the signal length is increased for the non-ideal case. In comparison to Fig. 3.4 and Fig. 3.5, the minimum W_1 metric is approximately 0.05 and 0.1 for the static and dynamic cases, respectively, indicating that the non-ideal case exhibits different behavior compared to the ideal case. Most notably, it can be seen that additional numerical errors exist in the dynamic frame cases compared to the static frame cases. For the dynamic frame results, the $N_s = 7.5 \times 10^6$ sampling time becomes a threshold that solidifies the frame size, similar to the observation in Fig. 3.4. However, interestingly, the poorer W_1 metric values beginning at $N_s \geq 7.5 \times 10^6$ indicate that the dynamic frame bounds of the non-ideal case differ from the reference solution. Specifically, the data of the largest sampling time ($N_s = 10^7$) exist within a range of $X \in [-20.31, 19.63]$ for the non-ideal case, while the reference data of the ideal case exists in the range $X \in [-20.29, 19.79]$. Thus, the frames of the two runs are different and increase the nonzero W_1 metric value due to frame misalignment. In comparison, the non-changing (static)

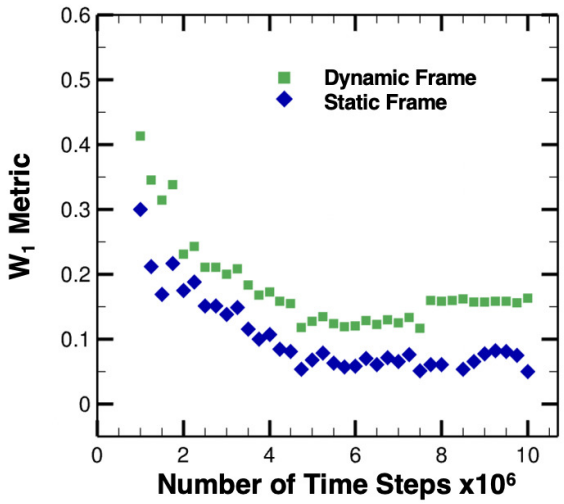

Figure 3.6: A comparison of W_1 metric values with various sampling times. The initial condition is set to $(x_0, y_0, z_0) = (10.01, 10.01, 10.01)$ and compared to the reference solutions obtained from $(x_0, y_0, z_0) = (10, 10, 10)$, which are the ones used in Fig. 3.4 for the *dynamic* frame and in Fig. 3.5 for the *static* frame, respectively.

frame is offset enough from the minimum and maximum bounds of the frame and the smoothed envelope better captures the reference solution, i.e., no numerical errors are introduced due to the change of the frame size and location. The results suggest the converged solution for the difference between the cases of $(x_0, y_0, z_0) = (10, 10, 10)$ and $(x_0, y_0, z_0) = (10.01, 10.01, 10.01)$ yields an W_1 metric of 0.05, shown from the static frame case. Additionally, comparing the results of the dynamic and static cases, the numerical errors due to the frame misalignment can be considered to yield a difference in W_1 metric ranging $0.05 - 0.1$.

Quantification of the choice of time lag and number of bins used in this thesis are also studied, although not shown at this time. The time lag can affect the W_1 metric of the system, but for the Lorenz attractors the difference is on the order of 10^{-2} between values for a time lag of 1 step or a time lag of 50 steps. For other systems, the choice of time lag may be more important in reducing the amount of data to be used while maintaining sufficient detail of the dynamics of the system. This will be studied further using a zero-dimension bulk plasma model in later sections. Conversely, the choice of number of bins used to generate the discretized PDF-TLPP

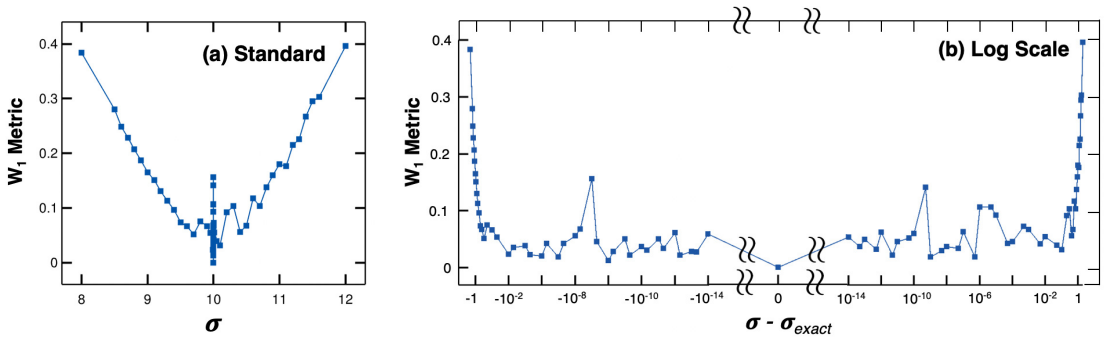

Figure 3.7: The response surface of the σ parameter for the near-solution region with $\sigma_{exact} = 10.0$ and $\sigma \in [8, 12]$. (a) The response surface that is monotonically decreasing until approximately $9.5 \leq \sigma \leq 10.5$. (b) The near-reference solution region showing the noisy W_1 metric values even with infinitesimal differences for a double-precision calculation. Note that the σ axis is displayed using a logarithmic scale to emphasize the small digit number calculations around σ_{exact}.

is fairly straightforward in that the more bins used, the more continuous the system is, but the longer computational time and the more data a single run will take to calculate the Wasserstein distance. Such user-defined parameters must be determined based on the problem of interest, but an approach based on maximizing the information content of the PDF-TLPP is currently being explored further elsewhere [64]. This will also be demonstrated using the zero-dimension bulk plasma model.

3.4 Sensitivity of the Constants (σ, β, ρ)

To best quantify the abilities of the optimizer, the sensitivity of the σ parameter is investigated. The range $\sigma \in [8, 12]$ is tested to determine the shape of the response surface of the Lorenz system that should guide the optimizer to the true solution. The input parameter values for β and ρ as well as the initial position (x_0, y_0, z_0) are held constant. The solution is embedded with a time lag τ of 20 steps to generate the TLPP and discretized by 50 bins in each direction to create the PDF-TLPP.

Figure 3.7 shows the W_1 metric results for $\sigma \in [8, 12]$ to illustrate the response surface of the Lorenz system prior to optimization. The overall smooth gradient of the W_1 metric shows that there are no local minima except for the near-reference solution region. Ideally, the optimizer will find the exact solution if the W_1 metric monotonically decreases toward the reference solution.

It can be seen in Fig. 3.7(a) for $|\sigma - \sigma_{exact}| > 0.5$, the W_1 metric is monotonically decreasing as σ approaches σ_{exact}. However, a highly stochastic structure can be observed in Fig. 3.7(b) for $|\sigma - \sigma_{exact}| \leq 0.5$. The Lorenz system is highly sensitive to infinitesimal changes in initial conditions. The results illustrate that small physical changes, even to the 15th decimal point (the minimum difference in a double-precision calculation), yield a large fluctuation in W_1 metric. This realization is indicative of three sources of errors, including (i) the discretization errors due to the numerical scheme chosen, (ii) the numerical errors due to frame misalignment when constructing the PDF-TLPPs, and (iii) the calculation of W_1 metric based on discretized PDF-TLPPs. This sudden highly stochastic region is likely caused by how costs add when distributions are close or overlapping versus far away. Especially when considering individual outlying trajectories as noise for these ideal, noiseless cases, fewer samples are required to "control" noise levels for calculations involving large distances compared to near-overlapping cases. While it is also important from the perspective of data sampling that a longer sampling time must be used to achieve statistical convergence, such numerical errors may grow or persist over time, particularly for a chaotic and non-periodic system. However, it must be noted that despite the numerical error in the vicinity of $\sigma \to \sigma_{exact}$, the minimum W_1 metric obtained by sweeping through parameters as shown in Fig. 3.7 is as low as $\sigma = \sigma_{exact} \pm 0.01$. The minimum W_1 metric achieved by the optimization process is discussed in the next section.

3.5 Automated Optimization

Having determined the behavior of the Lorenz system, this section presents the results from the automated optimization to evaluate the final optimized input parameters. The algorithm searches either for a specific convergence tolerance, the difference in minimal objective function values between generations, of $\varepsilon_0 = 10^{-7}$ or until the maximum number of function evaluations (user-defined) is reached. The convergence tolerance means that when the difference in optimal objective function values from generation to generation stagnates, the program terminates as the minimal objective function is considered located. The final results of any run can be used to inform new runs with narrower parameter ranges. This series of testing was performed using the Galaxy Simulation

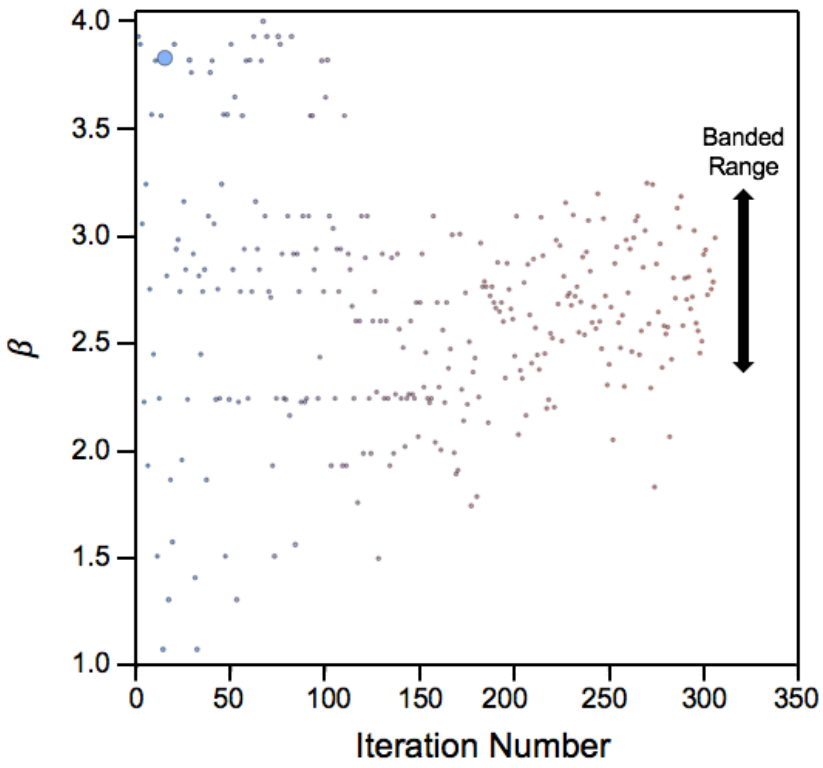

Figure 3.8: An example of how the evolutionary algorithm narrows the search range for a given parameter over the optimization process. For this figure, the β term is being optimized.

Builder program.

Having used narrow search ranges thus far, an initial range of 1-100 is set for the constant σ, and eventually all three constants, to gain an understanding of how the optimizer handles such a large range. Because of how the evolutionary algorithm employs a user-specified mutation rate as it generates new test values for the unknown input parameter, the range of possible values is reduced by some fractional amount but often does not reach a single optimal value for such large ranges. If one plots the value of the tested parameter as a function of iteration number, the reduction in search range becomes evident as seen in Fig. 3.8. The size of the points in Fig. 3.8 and any figures generated from the GSB program directly corresponds to the W_1 metric values (objective function) on the y-axis while the color of the points reflects the iteration number of each point (blue to red, chronologically) in the data-driven model. This new range can be used in a new optimization

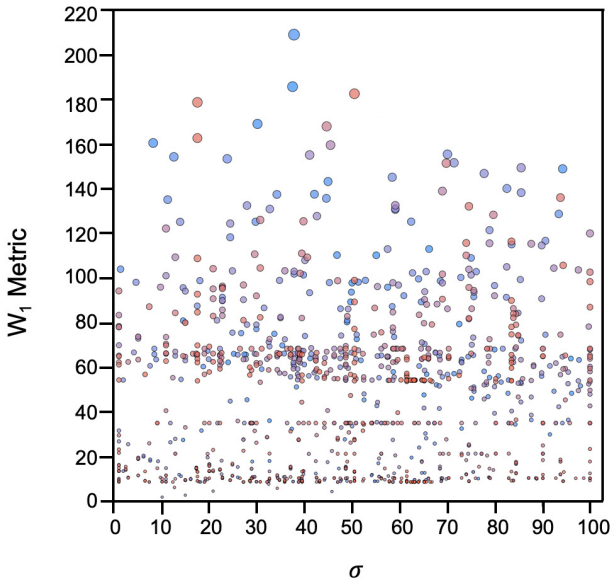

Figure 3.9: The initial overall response surface of the W_1 metric for the σ constant.

process to more thoroughly search the narrowed range. Note that the initial optimization runs are performed using a dynamic frame method due to the vast fluctuations in TLPP bounds. Once the range is reduced by at least 60%, the optimization scheme is reconfigured to use the static frame method to ensure better minimal W_1 values. Using this method, the initial range of 1-100 is searched for an average of five generations (sets of 32 points) before meeting the convergence tolerance. After narrowing the search range, the algorithm runs until the user-defined maximum number of iterations is reached since the tolerance level is rarely, if ever, met.

The reference solution uses an initial condition of $(x_0, y_0, z_0) = (10, 10, 10)$ with the exact input parameter values of $\sigma = 10.0$, $\beta = 2.667$, and $\rho = 28.0$ for a sampling time of $N_s = 10^7$ steps. The PDF-TLPP is created using a time lag τ of 20 steps and 50 discrete bins in each dimension. Every run uses an evolutionary algorithm with a population size of 32 points and a concurrency of the same, with 16 replacements per generation. The first result studies the results for only searching for one of the constants of the Lorenz problem, while the second search for all three constants simultaneously.

Figure 3.9 shows the chaotic and poorly defined response surface of the W_1 metric, the y-

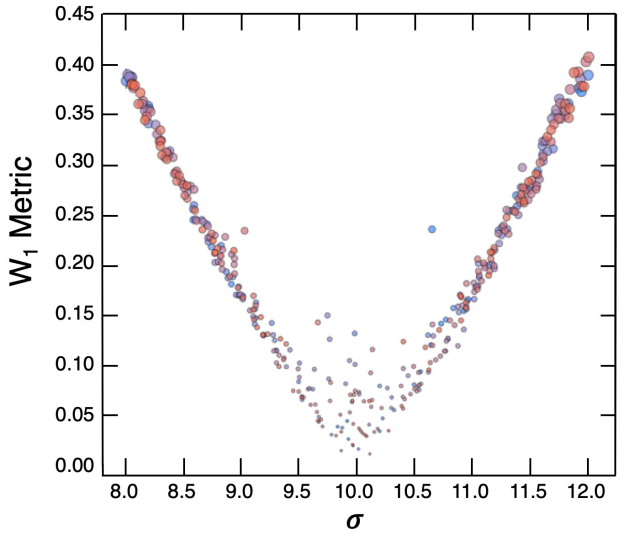

Figure 3.10: The trial output data during the optimization process obtained from the Galaxy Simulation Builder (GSB). Every tested σ value and its corresponding W_1 metric are shown. The point size represents the W_1 metric value, while the color represents the iteration number from blue to red, chronologically.

axis, for the full range of possible values given to the optimizer. This image gives evidence as to how the optimizer is unable to find a reasonably optimal solution for such a large range. If one uses the banded search range that becomes evident in the data, such as that in Fig. 3.8 to restart the optimization process, something like Fig. 3.10 is obtained. Figure 3.10 shows the response surface output from the GSB that directly corresponds to Fig. 3.7(a). This visual output shows the optimization run in a range of $\sigma \in [8.0, 12.0]$, which is the third optimization run, having reduced the data range twice. The results of the trial outputs during optimization agree with those shown in Fig. 3.7(a) but run for a CPU time of under 3 hours to generate approximately 500 results. At $|\sigma - \sigma_{exact}| > 0.5$, the response surface depicts a smooth gradient towards the exact solution, suggesting that optimization occurs towards the region of $9.5 < \sigma < 10.5$. Within this region, the trial outputs are noisier as σ approaches 10, which is consistent with the stochastic regime in the near-reference solution region in Fig. 3.7(b). Similar studies are performed on the two other parameters and initial condition, as well as 2D parameter sweeps, with similar results.

Table 3.1 depicts the numerical results of the optimization studies performed searching for the

Table 3.1: Optimization of σ while keeping $\beta = \beta_{exact}$ and $\rho = \rho_{exact}$. The reference solution is when $\sigma_{exact} = 10.0$, $\beta_{exact} = 2.667$, and $\rho_{exact} = 28.0$. Note that the relative errors $\varepsilon_\sigma = (\sigma - \sigma_{exact})/\sigma_{exact}$ are shown in parentheses.

	Dynamic	Static		
		Run 1	Run 2	Run 3
σ range	1.0 - 20.0	1.0 - 20.0	8.0 - 12.0	9.6 - 10.7
Optimal σ	9.711	9.843	10.104	9.973
(ε_σ)	(-2.8%)	(-1.6%)	$(+1.0\%)$	(-0.3%)
W_1 metric	**0.0995**	**0.0269**	**0.0110**	**0.0124**

optimal σ value while keeping β, ρ, and the initial conditions constant. The best result using the dynamic frame method is compared to three static frame tests with different σ search ranges. The relative error of σ is given by $(\sigma - \sigma_{exact})/\sigma_{exact}$. The dynamic case produces an W_1 metric that is close to 0.1, with a corresponding σ value approximately 3% different than the reference value. The reference solution for the dynamic optimization run lies within a frame of $X \in [-20.29, 19.79]$ while the best dynamic solution fit within a frame of $X \in [-20.21, 20.2]$, which means that this result includes a numerical error due to the difference in frame location and bin size between trial and reference solution. Significantly smaller W_1 metric values are obtained for the static frame runs, particularly when narrowing the search range for σ so that the GSB can try more data points in the vicinity of the exact solution. Two of the three tests, Run 2 and Run 3, result in a remarkable W_1 metric of approximately 0.01, with σ values less than or equal to a difference of 1% from the reference solution, i.e., $9.9 < \sigma < 10.1$.

Table 3.2 shows the results of the optimization for all three input parameters, e.g., σ, β, and ρ. The optimal parameters obtained, the relative error with the exact values, and the W_1 metric values are shown. Here, the total relative error accounting for all three variables is also calculated as $\sqrt{\Sigma\varepsilon^2} = (\varepsilon_\rho^2 + \varepsilon_\sigma^2 + \varepsilon_\beta^2)^{1/2}$. Two main results that can be observed are that the W_1 metric values and relative errors are much smaller for the static frame cases than for the dynamic frame case and that the static frame results are converging to a solution near the exact input parameters,

Table 3.2: Optimization of the constants σ, β, ρ for $\sigma_{exact} = 10.0$, $\beta_{exact} = 2.667$, and $\rho_{exact} = 28.0$. Note that the total relative error is given by $\sqrt{\Sigma\varepsilon^2} = (\varepsilon_\rho^2 + \varepsilon_\sigma^2 + \varepsilon_\beta^2)^{1/2}$.

	Dynamic	Static	
		Run 1	Run 2
σ Range	9.33 - 10.67	9.5 - 11.0	9.6 - 10.7
Optimal σ	10.331	10.107	10.051
(ε_σ)	$(+3.3\%)$	$(+1.1\%)$	$(+0.5\%)$
β Range	2.49 - 2.84	2.55 - 2.75	2.6 - 2.67
Optimal β	2.639	2.66	2.665
(ε_β)	(-1.0%)	(-0.3%)	(-0.1%)
ρ Range	25.13 - 29.87	27.0 - 28.2	27.5 - 28.2
Optimal ρ	28.180	27.983	28.056
(ε_ρ)	$(+0.06\%)$	(-0.06%)	$(+0.2\%)$
$(\sqrt{\Sigma\varepsilon^2})$	(3.4%)	(1.1%)	(0.55%)
W_1 metric	**0.0963**	**0.0200**	**0.0182**

$\sigma \rightarrow \sigma_{exact}$, $\rho \rightarrow \rho_{exact}$, and $\beta \rightarrow \beta_{exact}$. The ability of the optimizer to find an W_1 metric value on the order of 0.02 by optimizing three parameters simultaneously continues to provide evidence of verification of the data-driven method. The dynamic solution exists within a frame of $X \in [-20.29, 19.95]$ while the static solution is again set at $X \in [-25, 25]$.

4. ZERO-DIMENSION PREDATOR-PREY MODEL

To initiate work with plasma-physics based models, a zero-dimension predator-prey model is investigated. The predator-prey model, also known as the Lotka-Volterra equations, include a pair of first-order nonlinear differential equations that describe the interaction between two entities in a biological fashion, i.e., that of a predator and prey in nature [79, 80]. The Hall effect thruster community has employed the predator-prey model to obtain an appropriate scaling of the ionization length to better understand ionization oscillations [81]. The model is assumed valid inside a simplified discharge channel section of a Hall effect thruster. The neutral inflow is considered a constant value, the neutral outflow is allowed to fluctuate, and the channel length is defined using the channel geometry. Assuming a finite-volume method where properties neglect spatial variation and the fluxes at the interfaces are modeled, the ion and neutral continuity equations can be written as

$$\frac{\delta N_i}{\delta t} + \frac{N_i U_i}{L_{ch}} + \frac{2 N_i U_{i,w}}{R_{\triangle}} = N_i N_n \xi_{ion}, \tag{4.1a}$$

$$\frac{\delta N_n}{\delta t} + \frac{(N_n - N_{int}) U_n}{L_{ch}} = -N_i N_n \xi_{ion}, \tag{4.1b}$$

where *N* and *U* are the number density and mean velocity, the subscripts *i* and *n* denote ion and neutral atoms, respectively, N_{int} is the number density of neutral anodes at the anode, $U_{i,w} = (eT_e/M_i)^{1/2}$ is the ion acoustic speed, L_{ch} is the channel length, and $R_{\triangle}$ is the channel width. Using ground-state xenon atoms for the calculation, the ionization rate coefficient ξ_{ion} can be written as

$$\xi_{ion} \approx [A T_e^2 + B \exp(-C/T_e)] \left(\frac{8 e T_e}{\pi m} \right)^{1/2}, \tag{4.2}$$

where $A = -1 \times 10^{-24}$, $B = 6.386 \times 10^{-20}$, $C = 12.13$, T_e is the electron temperature, and m is the mass of a xenon atom.

The electron temperature is an implicit function of time and serves as the excitation input to the predator-prey system, creating a dynamic environment. This model is used to capture the ion oscillations generated by the ionization source before being accelerated out of the domain. This type of simplistic model is useful for understanding the temporal evolution of electron dynamics as a first step towards capturing the dynamic phenomena related to anomalous electron transport. The electron dynamics strongly influence the ion and neutral number density and ultimately affect thruster performance. For this system, the ionization rate coefficient is the key variable of interest as it is the connection between electron temperature and number densities as well as being a function of the thruster design and performance parameters. These parameters include the wall geometry, finishing materials, surface roughness, heating element employed, ambient chamber pressure, and nozzle characteristics. While 1D, 2D, and even 3D models of plasma discharge dynamics capture these details with higher fidelity, the bulk behavior of the system can be understood from the 0D discharge model of Eqs. 4.1a and 4.1b.

This work is intended to study the abilities of the calibration method with various testing conditions for a system of varying complexity. The work that forms the basis of this study discusses the relation of ionization oscillation damping to perturbations in the electron temperature. A simple non-constant model for electron temperature is a summation of sinusoidal waves of varying frequency and amplitude about some set average value. For this work, the electron temperature equation is written as

$$T_e = a_0 + \sum_{q=1}^{n} a_q \cos(\omega_q t), \tag{4.3}$$

where n is an arbitrary number of sinusoidal functions and t is the current time in the simulation. This study will use a one-, two-, five-, and ten-mode electron temperature equation for optimization to test varying qualities of the method. The one- and two-mode equations are considered the discrete modes while the five- and ten-mode are the continuous modes. The equations are solved using a simple forward Euler method with a time step of $1\mu s$ for 10^5 steps, an initial ion density of 10^{17}, an initial neutral density of 10^{19}, and the configuration of a Stationary Plasma Thruster

(SPT)-100 ML thruster [82] as in work by Hara *et. al.* [81]. An average electron temperature of 20 eV is set and if not stated otherwise, the time lag is set to 150 steps while 100×100 bins are used to generate the PDF.TLPP. The plots of ion versus neutral number density are not time-lagged.

This work will attempt to optimize the electron temperature equation by using either the time-embedded ion number density time evolution or both the ion and neutral number density output signals mapped against each other. These tests are intended to better emulate some differences that may occur between computational and experimental data. Testing the convergence properties of the optimization method for different time lags and noise, different sampling rates, simplified models being optimized to a continuous solution, having gaps in the data, and how dense the PDF-TLPP must be are all subjects of tests in this section.

4.1 Time Lag Dependence and the Effect of Noise

As previously mentioned, the choice of time lag is significant to the optimization process as it may hide correlations or noise, or show dependencies that do not truly exist. The resulting image can also either help or hinder the optimization process depending on the generated image. The more unique the image, the better the resulting optimization run will be. Figure 4.1 shows the same single mode system embedded with time lags varying from 50 to 600 time steps in size. These specific values were chosen because the period of the single-mode sinusoidal function lasts for 628 time steps before repeating. Not pictured is the straight-lined diagonal plot that results from using either 0 or 628 time steps for the time embedding. The variety in PDF-TLPPs show how the time lag affects the optimization process. One of the characteristics of an unique image is a lack of overlapping signal lines such as those in the 300 and 350 time step cases. Also any images that seem to have very small regions of high intensity weights are undesired as those regions of high weight will skew the optimization process. Thus time lags such as 100, 150, or 200 are the most desirable embeddings for this specific system.

The inclusion of high level noise to the system, implemented as an additional higher frequency sinusoidal function, creates an entirely different image, seen in Fig. 4.2. Noise is undesired in the system and creates many more instances of overlap in signal traces, thus making the selection of an

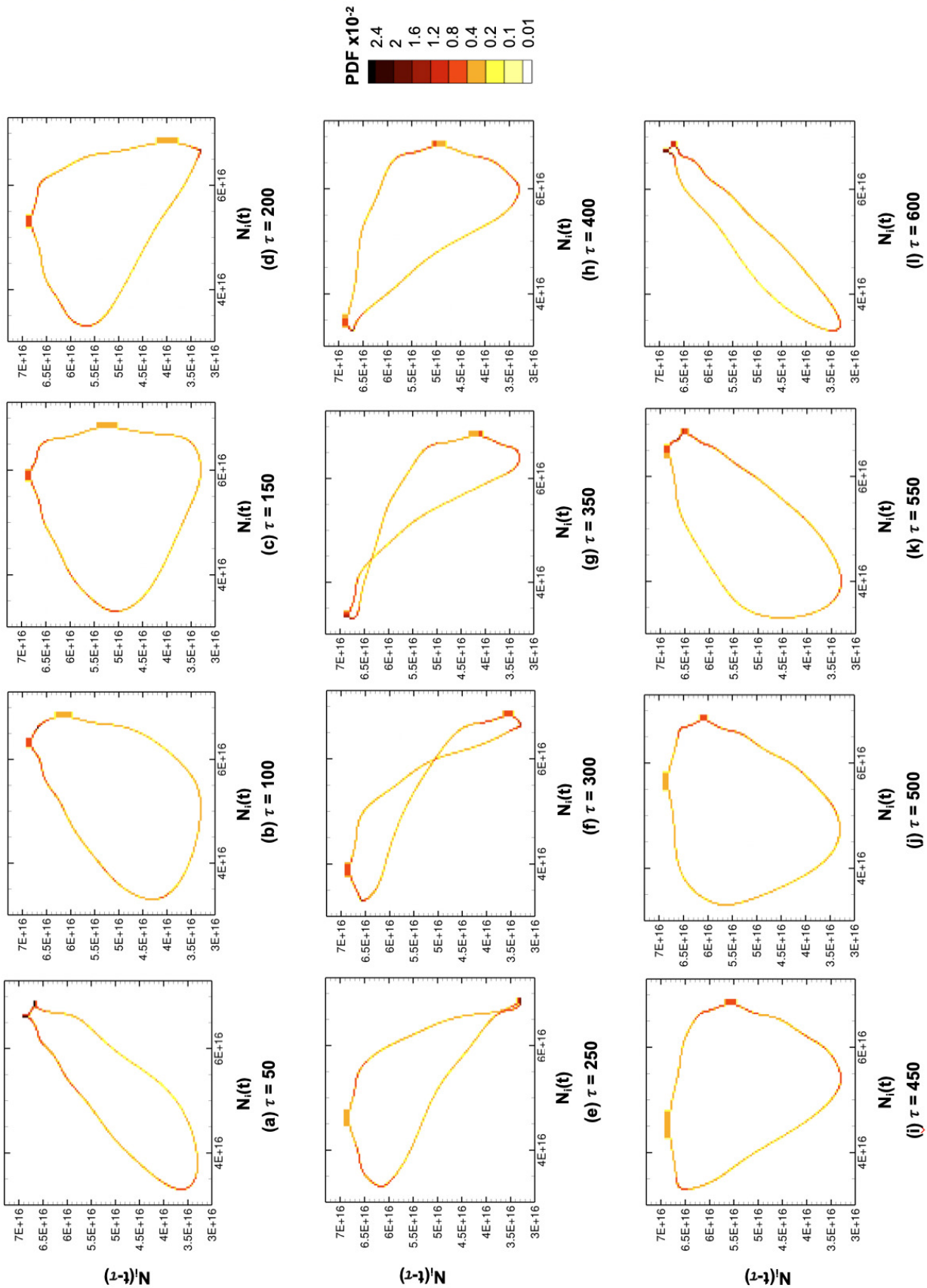

Figure 4.1: The effect of the time lag on the appearance of the system in the phase space portrait.

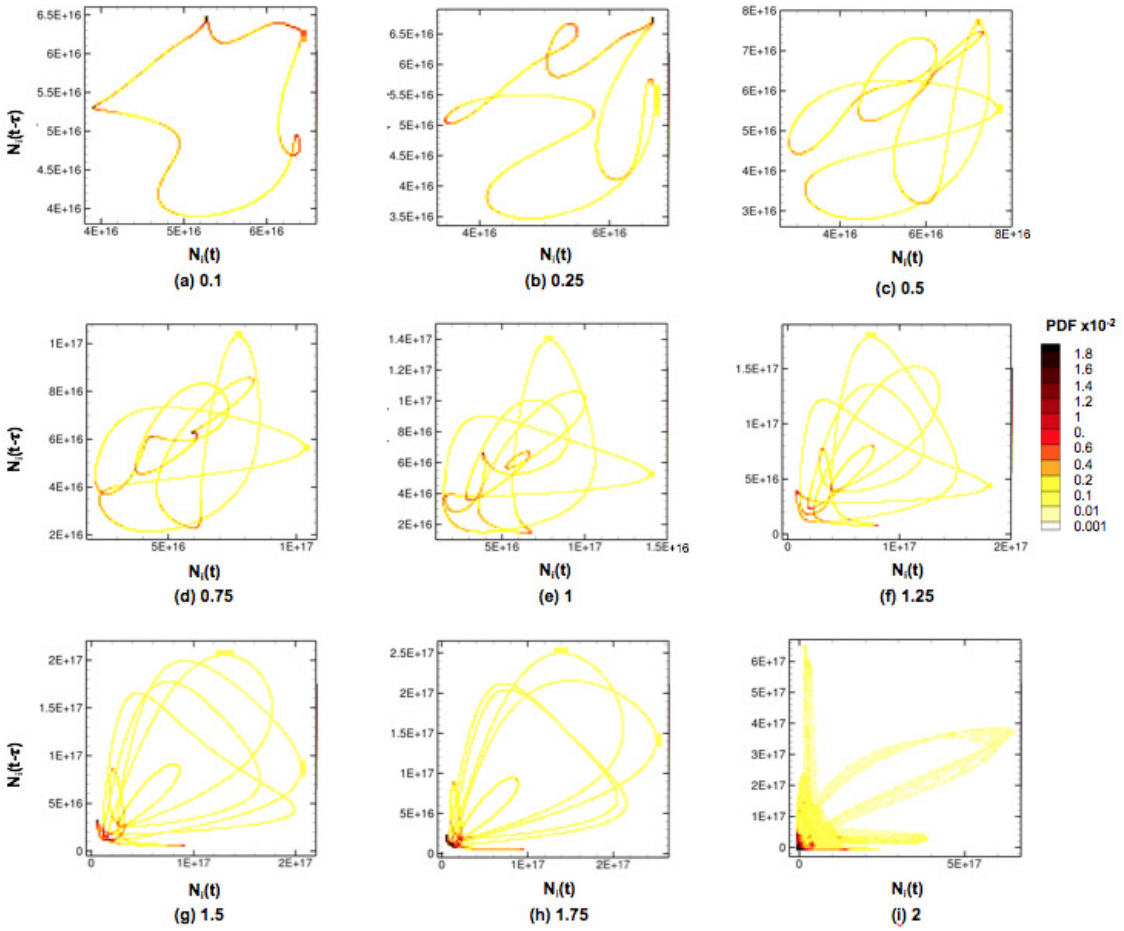

Figure 4.2: The effect of the noise on the appearance of the system in the phase space portrait. A 40,000Hz signal is applied to the original one-mode signal. The values under each figure correspond to the amplitude of noise applied to the system.

appropriate time lag more difficult. The inclusion of noise may be one of the stronger requirements for more than one delay embedding to be used to generate a unique mapping. Even the amplitude of the noise can significantly affect how much overlapping exists within the data as shown in Fig. 4.3. Though noise is not specifically studied in this work, the ability of the optimizer to overlook "noise" is examined by using a lower-mode solution to estimate a higher-mode solution.

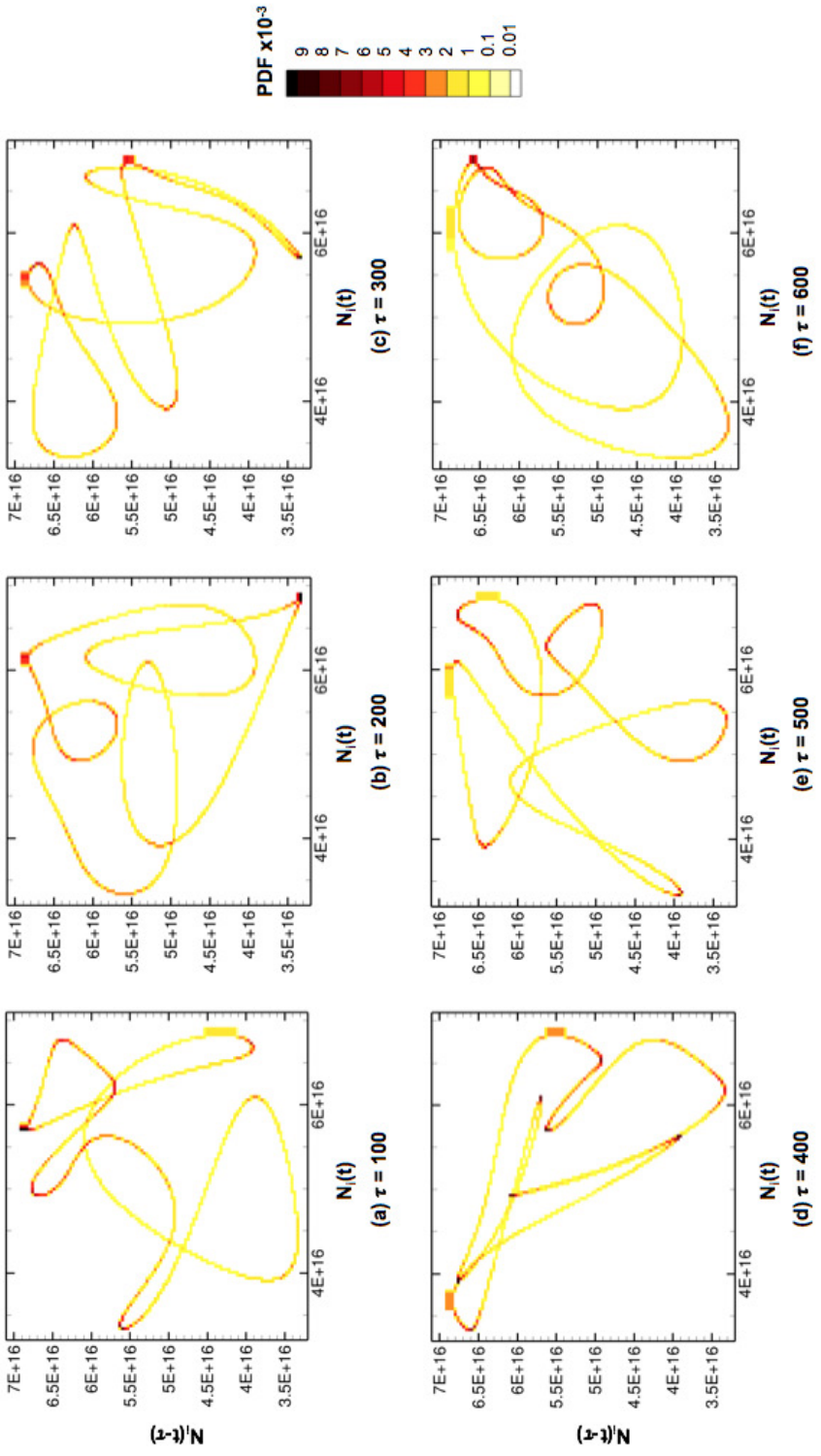

Figure 4.3: The effect of the time lag on a system with a constant noise on the appearance of the system in the phase space portrait.

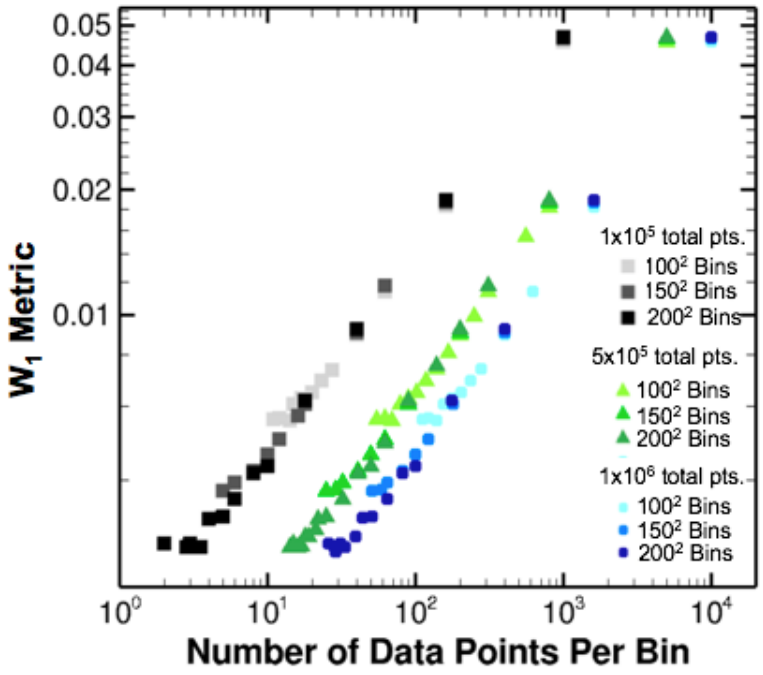

Figure 4.4: The effect of the density of the PDF-TLPP, or the number of bins per direction of the PDF-TLPP, on the W_1 metric.

4.2 PDF-TLPP Density

Different grid sizes affect the W_1 metric because of how distances and weights become differently distributed. If the data were discretized as one singular bin, the W_1 metric would always be zero so long as the TLPP frames remain the same. Note that this occurs because the geometric center of each bin is used to track the location of the bin. If instead the center of mass within each bin is used to track the location, one would obtain a more unique mapping of the system. This more robust method is left for future work as it induces a bias error to the W_1 calculation in addition to the variance error already in existence. Using geometric bin centers and the static frame methods set a noise floor on the W_1 metric expected for finite data and a given mesh resolution. Thus, a study of the optimal discretization of the TLPP is performed in terms of the density of points per bin (alternatively the number of bins per direction) in the PDF.

The first convergence study examines the significance of the average number of data points per

PDF bin and its effect on the W_1 metric value. Fig. 4.4 shows this exact correlation for different sized datasets and reference solutions using three different sizes of grids. The positive slope in this figure is caused by a larger number of points per bin corresponding to a smaller number of bins in the PDF-TLPP. Thus, as the number of points per bin decreases and the PDF-TLPP becomes more similar to the reference solution, the lower the W_1 value becomes. The asymptotes of each shade of color indicate where the reference solution exists and how close to zero the optimal solution is expected to be if a non-ideal solution is generated. The similarity between graphs indicates that the size of the reference dataset is irrelevant, only the discretization of the TLPP seems to affect the W_1 metric. Here the slope is unitary because of the difference in bias versus variance errors. Using a static frame makes the bias errors identical so that they do not contribute to the W_1 metric even though they would be non-zero. Increasing the number of points per bin to infinity drives the variance errors to zero, but finite bin size still generates a non-zero bias error. For example, though the number of bins used in the trial and reference solutions differ, the W_1 metrics are similar because the bias error dominates the variance error caused by the differently sized bins. This occurs regardless of how many bins constitute the reference solution, seen in Fig. 4.4, providing evidence of this source of error. If it were computationally possible to drive the PDF-TLPP to having only zero or one points per bin, almost every weighted bin would likely have to be moved some finite distance, causing variance errors to finally overtake the bias errors.

4.3 Subset Convergence

The next test performed is used to study the convergence of data while emulating a case where not all of the data is available for optimization. This is accomplished by taking the reference solution TLPP and using a MATLAB function to take a random subset of user-defined size. The same size subset can be created more than once, and the resulting vector will be different due to the randomizing function. Thus, multiple tests of the same subset size are performed to obtain an average and standard deviation value for each size. This is performed before the solution is binned as a probability distribution function. Depending on the size of the subset taken, the weight of each data point is artificially altered such that the full-sized reference solution and subset retain the same

weight. As an example, if half of the data points are used to form the subset, the weight of each data point becomes two instead of one. This is done to prevent additional numerical errors from being introduced during this study. Note that the subset is taken after the time embedding so that the time lag is not affected by the subset and does not induce unrealistic errors into the process.

This subset study was performed for three different sizes of reference solutions, using 100,000, 500,000, and 1,000,000 data points. To obtain some range of standard deviation, each subset size was randomly generated five times to obtain some range of possible values. As can be seen in Fig. 4.5(a), the data appears to exist within an almost constant banded height regardless of the subset size. The average values are shown in Fig. 4.5(b) for all three sizes of reference solution. The standard deviation for each subset size was calculated to be below a value of 0.0001, which is of small consequence for the W_1 metric values obtained and so do not appear in Fig. 4.5(b). What is significant to note, though, is that all three reference solution sizes share a common linear slope of -0.5 in the logarithmic plot. As the reference solution increases in number of data points, the converged W_1 metric is able to be lowered compared to the 100,000 point reference solution. This gives evidence to the need to use more data points as opposed to fewer to help lower the minimum W_1 metric and in turn improve any gradient in the response surface that may exist. It is noted that the converged W_1 metric still experiences difficulty in reaching a value lower that 0.001, which may be a limitation on the system for all but exact solution W_1 values.

4.4 Number of Sinusoidal Periods

A study is performed to understand how long of a simulation is needed for the difference in signal length to be rendered minimally significant. For this study, an extra half period of the simulation is added to the trial solution compared to the reference solution. For instance, while the reference solution may consist of 16 full periods, the trial solution will contain 16.5 periods. The number of full periods per run is increased to examine the convergence in W_1 metric in Fig. 4.6.

Few test cases were run for this study as the trend is clear. As the number of full periods increases, the effect of the extra half-period diminishes to very low W_1 metric values. A half period is only 0.05% of the largest dataset of approximately 1000 full periods and is one of the few

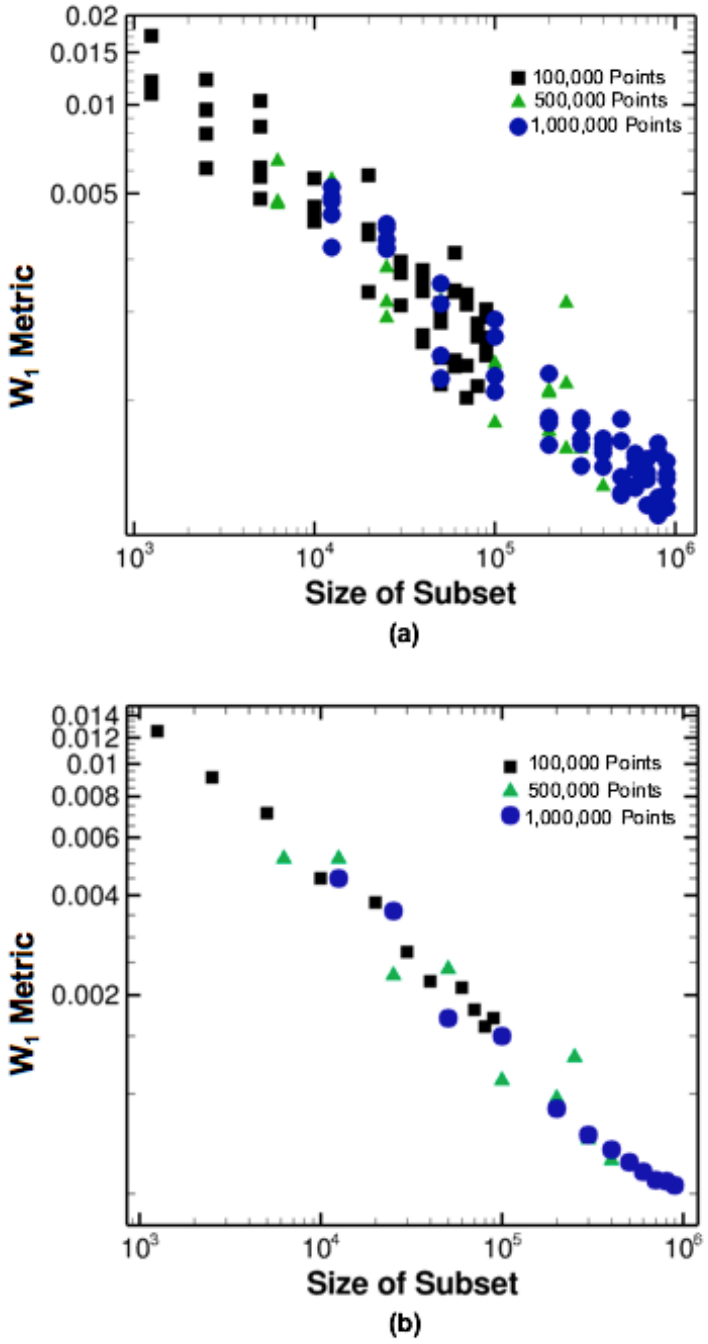

Figure 4.5: (a) The change in W_1 metric for various sized, randomly selected subsets of data for a non-exact solution. Three different sized reference solutions are used in this figure to determine what affect the size of the reference solution has on the minimum W_1 value. Each size subset was tested five times to determine the size of the range for all subset sizes. (b) The change in W_1 metric for various sized, randomly selected subsets of data for a non-exact solution shown as the average of each of the five tests.

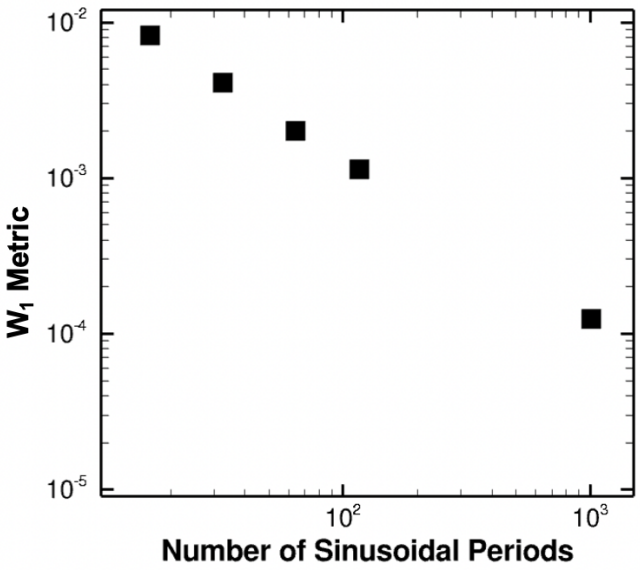

Figure 4.6: The change in W_1 metric for different length simulations when an extra half period is added to the trial solution.

cases in this work to yield a W_1 value less than 0.001. The slope of -1 is likely due to a uniform bin bias error that arises from the 0th order uniform bin quadrature. Maintaining the same bin size keeps this bias error a constant, but nonzero, value. The greater number of data points in each bin relative to the number of data points caused by the extra half-period of the trial solution is a large enough ratio to lessen the effects of the extra data. Thus, even if a simulation and experimental results are not exactly identical in run time, their data can be compared if the difference is a small enough fraction of the total signal length.

4.5 Discrete vs. Continuous Testing

The PDF-TLPP of this predator-prey model varies greatly depending on the number of sinusoidal functions used in the system. Figure 4.7 shows a one-mode, two-mode, and ten-mode electron temperature signal. The reference equations are as follows:

$$T_e = 20 + 3\cos(10000t), \tag{4.4a}$$

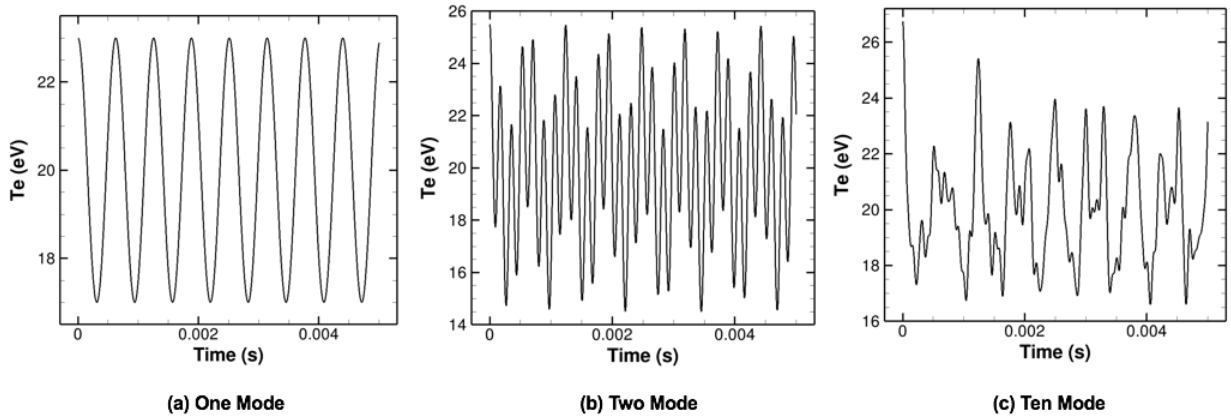

Figure 4.7: Continuous and discrete electron temperature equation signals using (a) one mode. (b) two mode and (c) ten mode temperature equations.

$$T_e = 20 + 2\cos(10000t) + 3\cos(35500t), \tag{4.4b}$$

$$\begin{aligned} T_e = 20 &+ 2\cos(10000t) + 1.5576\cos(12125t) + 1.2131\cos(15000t) + 0.9447\cos(21000t) \\ &+ 0.7358\cos(25000t) + 0.5730\cos(36000t) + 0.4463\cos(40000t) \\ &+ 0.3475\cos(45678t) + 0.2702\cos(62750t) + 0.2108\cos(75000). \end{aligned} \tag{4.4c}$$

As more modes are added, with decreasing amplitudes to emulate exponential decay, the solution changes. Each of these solutions will be used in this section to study the ability of the calibration method to handle discrete and continuous test functions simultaneously. Studies were performed to attempt using different discrete and continuous solutions together. This emulates attempting to resolve highly complex, multi-mode experimental data using far fewer modes in a computational model. Though additional effort should be made to determine an optimal number of discrete modes needed to capture the dynamics of a continuous function, this work shows that the optimizer is capable of recognizing trends and significant modes in the system.

The discrete versus continuous testing is performed in a matrix-type style. The four genres of

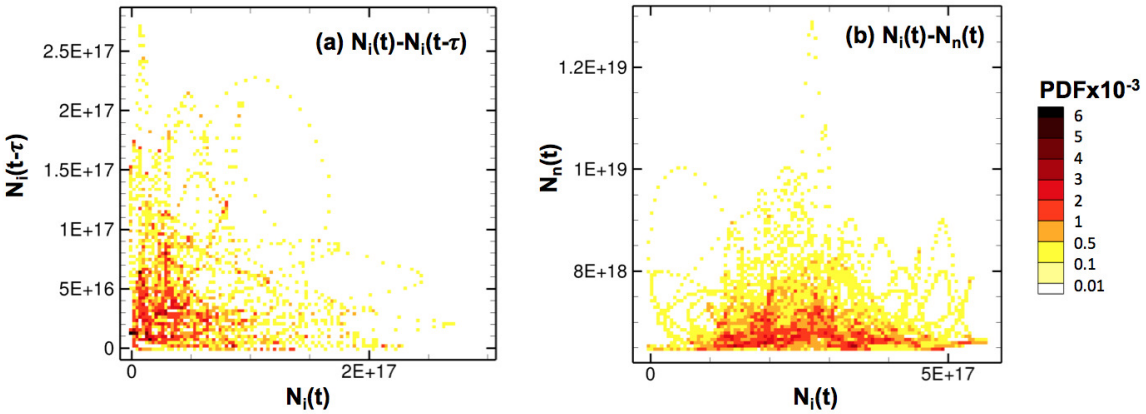

Figure 4.8: (a) The ten mode system with a phase portrait built by taking a time lag of the ion number density. (b) The ten mode system with a phase portrait built by comparing ion and neutral number density.

test cases are: (i) discrete-to-discrete, (ii) discrete-to-continuous, (iii) continuous-to-discrete, and (iv) continuous-to-continuous. For each test case, a variety of options are used for the studies. One studied effect is the use of either the ion number density embedded to a higher dimension by taking a time lag or simply plotting the ion number density versus the neutral number density. An example of how using the ion number density or both the ion and neutral number density affect the appearance of the system in phase space are seen in Fig. 4.8. Each of the four cases are shown in the figures below, with different test cases represented by different colors to show a collective understanding of how well the optimizer functions for these different cases.

Figure 4.9(a) shows the result of discrete-to-discrete testing. Specifically, a two-mode reference solution is attempted to be found by a two-mode trial solution. The different test cases include using both the time-lagged ion number density embedding and the ion versus neutral number density plot. Searches were performed when (i) the frequencies are set and the optimizer only searches for the amplitudes, (ii) the amplitudes are set and the optimizer searches for the frequencies, (iii) both frequencies and amplitudes are searched, and (iv) all values including the initial average temperature value are searched. The reference solution is plotted in black. Immediately noticeable is the optimizer's ability to find relatively well-matched solutions for the larger-amplitude frequency

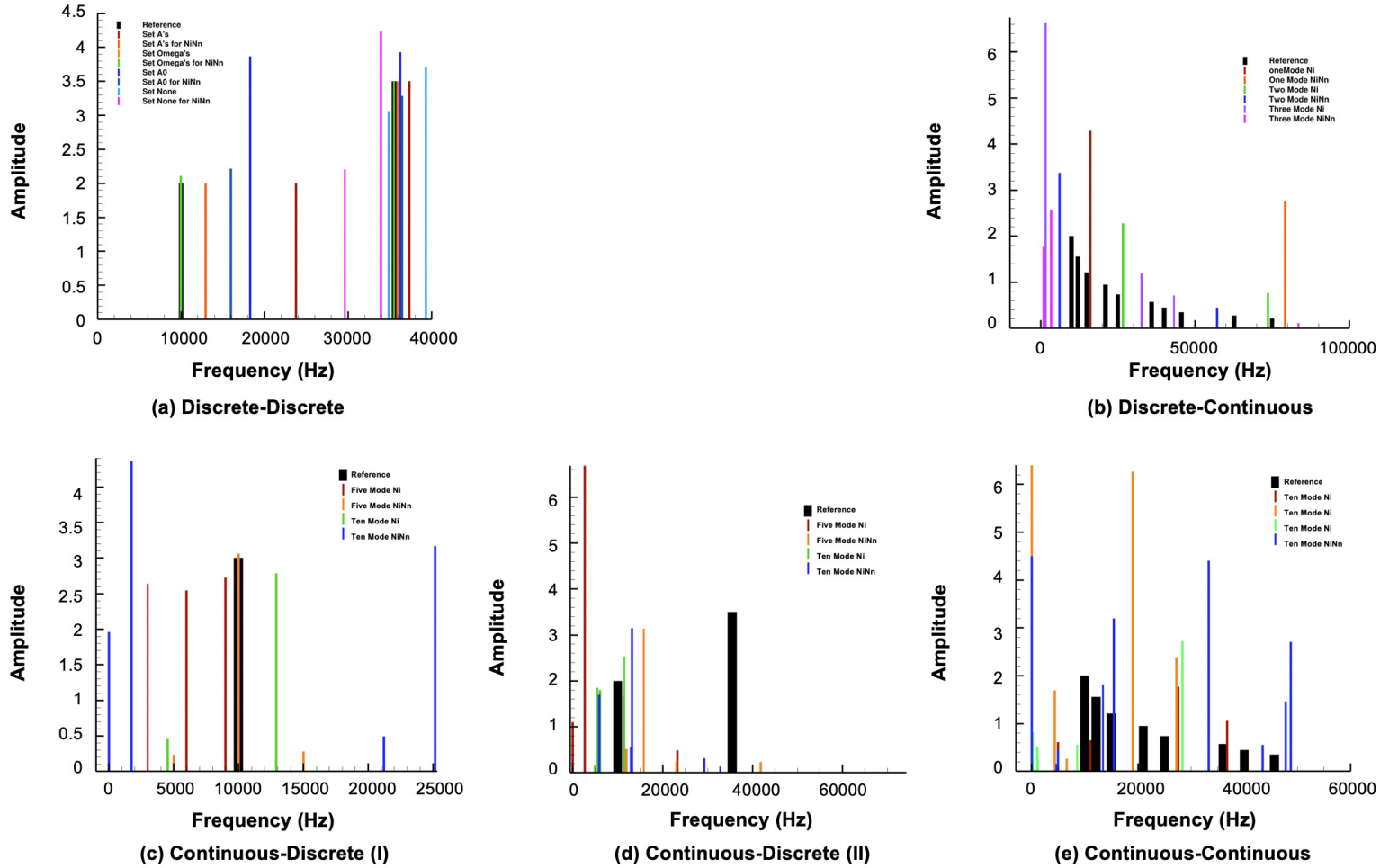

Figure 4.9: The results of the discrete versus continuous testing. (a) A discrete, two-mode solution as optimized by one-, two-, and three-mode calibration equations using two different phase portraits. (b) A continuous, ten-mode solution as optimized by one-, two-, and three-mode calibration equations using two different phase portraits. (c) A discrete one-mode solution as optimized by five- and ten-mode calibration equations using two different phase portraits. (d) A discrete two-mode solution as optimized by five- and ten-mode calibration equations using two different phase portraits. (e) A continuous ten-mode solution as optimized by a ten-mode calibration equation using two different phase portraits.

Table 4.1: Comparison of the W_1 metric values for the discrete-discrete mode testing for the time-embedded ion number density PDF-TLPP and the ion versus neutral number density PDF-TLPP.

	$N_i(t)$ vs. $N_i(t-\tau)$	N_i vs. N_n
Set Amplitudes	0.0894	0.3894
Set Frequencies	0.0337	0.3783
Set Average Value	0.0700	0.6598
Set Nothing	0.0958	1.4564

for almost every test case. Similarly, though the frequencies are not as matched, the amplitudes for the lower-frequency function are well-matched in many cases. Any missing colors or lines indicate that the optimizer gave that frequency a low or zero value such that they are not visible on the plot. The optimizer's ability to find the dominant frequency of the system proves a promising result for this low-frequency study. In comparing the time-embedded ion number density plot to the ion versus neutral number density plot, the time-embedded PDF-TLPP consistently yields better W_1 metric values. The following table depicts the differences minimal values obtained by each type of plot.

It is immediately evident from Table 4.1 that the time-embedded PDF-TLPP performs better than plotting two signals versus each other. This result is seen in all four of the test cases shown in this section, and so only this case is used for demonstrational purposes. Most likely, the neutral density reacts too slowly in the system compared to the ion density, meaning it does not provide a good representation of the dynamics. Conversely, the time-embedding uses the dynamics of the ion density against itself and avoids this issue.

Figure 4.9(b) shows the discrete-to-continuous testing. A ten-mode reference solution is used,

but only one-, two-, and three-mode trial solutions are allowed to be optimized. Here both the frequency and amplitude of the discrete-mode simulations are searched. Despite the difference in number of available modes, the solutions once again show that the optimizer is capable of understanding the general trend in data. For nearly all cases, the biggest peak of data occurs near the original solution's peak with lower values following. Because fewer modes are used to construct the optimal solution, their amplitudes are larger as they look at the summation of the amplitudes of all the other modes. Equivalent results are seen for both methods of TLPP creation. Using fewer modes to replicate a multi-dimensional mode emulates attempting to determine dominant modes in a noise-filled data set. The ability for the optimizer to recognize that the largest amplitudes existed in the lower-value frequencies shows that the optimizer is capable of appropriately weighting modes of differing importance. While not perfect, these results are indicative of ways to cope with noise, such as finding a way to reduce the amplitude, and therefore significance, of the noise frequencies so that the optimizer focuses on the larger-amplitude modes of the system.

Figure 4.9(c) and (d) show the continuous-to-discrete testing. This test uses both a five- and ten-mode trial solution to match a one- and two-mode reference solution with each reference solution plotted separately. While most test cases are able to determine a handful of zero-valued amplitudes, meaning that the optimizer recognizes not all of the modes are needed to replicate the reference solution, the results leave room for further improvement. For these cases specifically, the optimizer experienced great difficulty in (i) finding sets of input parameters that did not cause the simulation to fail and (ii) finding solutions with W_1 values below 5. The optimized solutions for the two-mode reference case saw W_1 metric values as high as 11. In the best cases, the W_1 metric values for these results are on the order of 0.2 which is far higher than would be expected based on previous work. Additionally, for this specific series of test cases, the W_1 metric value for the ion versus neutral number density plot tests are noticeably worse than those of the ion number density embedded dimension.

Figure 4.9(e) shows the result of continuous-to-continuous testing. This is performed both for a five-mode trial solution and ten-mode trial solution with a ten-mode reference solution in both

cases. Due to the complexity of the system, as well as the complexity of the PDF-TLPPs seen at the beginning of this chapter, these results are not as optimal as the discrete case. Nonetheless, what is significant about these results are that the optimizer is able to recognize that the dominant frequencies inhabit the lower range of values . This is especially noticeable for the five-mode test case where all results lie below 40,000Hz. Moreover, the optimizer is able to recognize a general downward trend after some peak value as the frequencies increase. The ten-mode solution is similar, though it often contained many zero-valued amplitudes as part of its results.

5. HPHALL TESTING[1]

For the Lorenz problem, as an example of a system consisting of few degrees of freedom, the 2D TLPP is shown to find near-optimal input parameters of the dynamic system. However, a nonlinear system with many degrees of freedom may exhibit more complex emergent dynamics. To demonstrate the abilities of the 2D TLPP, the data-driven model calibration approach is used to investigate its applicability for a HET discharge plasma.

5.1 A Hall Effect Thruster Model

The selected plasma discharge model, HPHall, is a 2D axisymmetric, hybrid particle-fluid Hall thruster model initially developed by Fife and Martinez-Sanchez [6] and further extended to several different versions[83, 19]. This numerical model simulates plasma behavior and plasma interactions during the operation of the thruster. A particle-in-cell (PIC) simulation is used for the heavy species while a fluid model is used for electrons. The electron fluid model utilizes a quasi-1D approach where the dynamics along and across the magnetic field lines are decoupled. Hence, a 1D cross-field transport equation for electrons based on the drift-diffusion flux is solved in the axial direction, represented by the thermalized potential. The Boltzmann potential relation is used to evaluate the potential profile in the direction along the magnetic field lines [21]. Due to the numerical fluctuations that occur from using computational macroparticles, various time-averaging techniques are used in HPHall. For instance, the electron momentum equation is solved assuming a quasineutral plasma at the ion time step, while the electron energy equation is solved using a sub-timestepping technique.

The particular internal phenomenon of interest in this paper is to investigate the effects of the anomalous electron transport across magnetic field lines [20]. Recent studies suggest that the cause of such anomalous transport may be due to plasma-wall interactions [10, 84] and electron cyclotron

[1]Reproduced from C.M. Greve, K. Hara, R.S. Martin, D.Q. Eckhardt, and J.W. Koo. "A data-driven approach to model calibration for nonlinear dynamical systems" *Jour. of Appl. Phys.* Vol. 125, No. 244901, 2019. with permission of AIP Publishing.

drift instabilities [26, 85, 8, 86, 87]. In reality, all such nonlinear dynamics may be closely coupled, causing difficulty for a physics-based model to achieve predictability [5]. Available experimental data such as discharge current oscillations [82] to ion velocity distribution functions [88, 89] and probe measurements [90], could be used as reference solutions for data-driven models.

For this work, HPHall simulates a generic model of sub-kW class HET run in an unoptimized configuration. While this model could be calibrated using experimental data as the reference solution, an arbitrary configuration of the computational model is selected as the reference solution to perform a code-to-code self-verification. Validation of the data-driven model against experimental data is reserved for future work, since verification is a necessary prerequisite to any validation study.

The crossed-field electron transport is described assuming a drift-diffusion flux for electrons, which can be written as

$$u_{e,\perp} = -\mu_{\text{eff}} \left[E_\perp + \frac{1}{en_e} \nabla_\perp (n_e k_B T_e) \right], \tag{5.1}$$

where $u_{e,\perp}$ is the electron bulk velocity in the cross-field ($\perp$) direction, μ_{eff} is the effective cross-field mobility, $E_\perp$ is the electric field, n_e is the electron density, k_B is the Boltzmann constant, T_e is the electron temperature, and e is the elementary charge. The electron flux is used in conjunction with the ion flux, considering a current balance: $\nabla_\perp(n_i u_{i,\perp}) = \nabla_\perp(n_e u_{e,\perp})$, where the ion density $n_i = n_e$ is assumed due to the quasineutral assumption and $u_{i,\perp}$ is the ion velocity. Here, the effective mobility is given by

$$\mu_{\text{eff}} = \frac{\mu_0}{1 + \Omega^2}, \tag{5.2}$$

where $\mu_0 = e/m_e\nu_m$ is the non-magnetized electron mobility, m_e is the electron mass, ν_m is the electron momentum transfer collision frequency, $\Omega = \omega_{ce}/\nu_m$ is the Hall parameter, $\omega_{ce} = eB/m_e$ is the electron cyclotron frequency, and B is the magnetic field amplitude. The anomalous electron transport is typically accounted for in the ν_m term, which can be considered as $\nu_m = \nu_{m,clas} + \nu_{m,ano}$, where $\nu_{m,clas}$ is the collisional (classical) contribution from intermolecular collisions and $\nu_{m,ano}$ is the contribution from the anomalous electron transport [5]. In HPHall, the anomalous

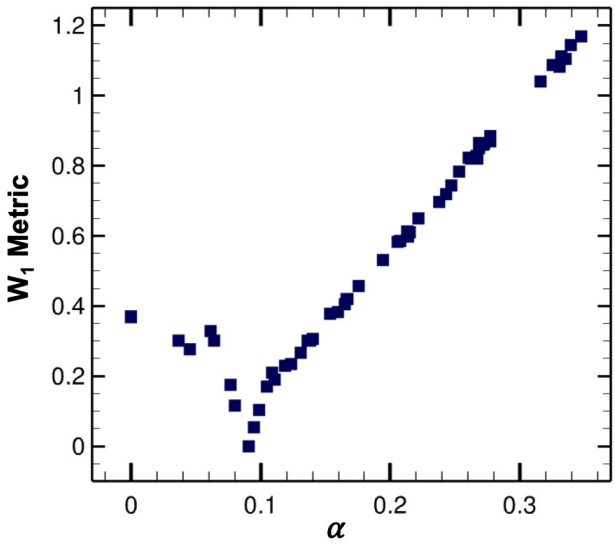

Figure 5.1: The W_1 metric response surface of the coefficient for the Bohm mobility model, α, when the reference solution is chosen to be $\alpha_{exact} = 0.0909$.

electron transport is calculated using the summation of four mobility models. These models, based on the work of Jorns [20], are calculated individually and summed using coefficients that describe the strength of each mobility model. For the sake of simplicity in this paper, only the Bohm mobility model is taken into account. Thus, the mobility model of HPHall can be expressed as

$$\nu_{m,ano} = \nu_{Bohm} = \frac{\alpha}{16}\omega_{ce}, \tag{5.3}$$

where α is the optimized coefficient describing the strength of the anomalous electron transport. It is often assumed that the anomalous transport contribution varies in different regions of the thruster channel, such as the near-anode, ionization, and acceleration regions [19], however, a uniform α is considered in this paper for simplicity. The other details of the HPHall is referred to in other literature [6, 83].

For further simplicity in this study, the wall conductivity model is deactivated. The effects of the chamber pressure are also turned off in this model. Only singly charged ions and associated ionization are taken into account. A nominal condition for the magnetic field strength, mass flow

rate, and discharge voltage is chosen. The number of macroparticles for ions and neutral atoms are approximately 60,000 and 100,000, respectively. The time step is 0.5 ns, and the electron sub-time step is 50 ps. The simulation is run to 2.5 ms for a converged solution.

5.2 W_1 Metric, PDF-TLPP, and Power Spectral Density

A code-to-code self-verification will be performed by constructing a reference solution from the data obtained from HPHall. Here, a reference dataset is constructed from the discharge current signal dynamics when $\alpha = 0.0909$, hence $\alpha_{exact} = 0.0909$. The TLPP of the discharge current is obtained for the reference solution and the GSB optimizer is configured to determine the value of α (the input parameter of interest) based on the reference case. All other variables of HPHall such as the discharge voltage, magnetic field strength, and collision model parameters are considered fixed inputs for all data-driven simulations to focus solely on obtaining the optimized mobility coefficient. The time lag τ is set to 150 time steps which corresponds to a physical time of 6 μs, based on work performed by Martin *et. al.* [64] which also explains in detail the application of Takens' Theorem to Hall effect thrusters. The PDF-TLPP is constructed using 50$\times$50 bins. Similar to the Lorenz problem, the effects of dynamic and static frames are investigated. While the Lorenz system optimization takes approximately 5 CPU hours for 1000 iterations, the HPHall simulation takes 2000 CPU hours for 500 iterations.

Figure 5.1 shows the response surface of the Bohm mobility model coefficient, α, as a gradient towards the reference solution, $\alpha_{exact} = 0.0909$. The limited search range is caused by the Bohm mobility model being the only model "turned on" in this configuration, resulting in only $\alpha \leq 0.4$ yielding successful HPHall runs for the given numerical inputs, e.g., number of particles, time step, and grid size. Even in this small range, the response surface has a gradient that can be determined by the optimizer. Though the same problems related to the W_1 metric and numerical inaccuracies exist, the general function of the optimizer can still be established similar to the Lorenz system.

Because the Hall thruster simulation presents a more defined set of resonant frequencies in its discharge current compared to the Lorenz system, a comparison of the power spectral density and the PDF-TLPP is shown at this point. Figure 5.2 shows the discharge current, PDF-TLPP, and

FFT of the reference ($\alpha_{exact} = 0.0909$) and off-reference ($\alpha = 0.3$) solutions. Immediately visible is the difference in time-averaged discharge current values between (a1) and (b1). The difference in the average discharge current is represented as a "distance" in the PDF-TLPPs, demonstrating that a longer distance between two images results in a larger W_1 metric while other norms (e.g. L_2) may not. In comparison, the power spectral density shows the time-averaged quantities as the lowest frequency mode. While the PDF-TLPP captures the overall structure of the time dependent signal, e.g. maximum, minimum, and spread of the signals, the power spectral density obtained by FFT directly shows the differences in the frequency. Although not shown in this paper, the W_1 values obtained from the Fourier-based approach exhibit a similar convergence property compared to PDF-TLPPs. One potential advantage of using PDF-TLPPs is for a real-time calibration of dynamic phenomena. Time-dependent reference data, say, acquired from real time measurements, can be updated seamlessly using PDF-TLPPs while the power spectral density must be recalculated for all discrete frequencies when adding data online.

5.3 Finding the Anomalous Transport Coefficient in HPHall from the Reference Data

Optimization of the anomalous transport coefficient is performed on the Hall thruster simulation using the data-driven calibration model. The optimization results are shown in Table 5.1. The optimization run by GSB uses the same evolutionary algorithm as used for the Lorenz system. The convergence criterion is set to $\varepsilon_0 = 10^{-7}$ and the maximum number of iterations is 200. The results of Table 5.1 are consistent with the results from the Lorenz system. While the W_1 metric value is again non-zero, the difference in trial and reference solution is well below 1% with a match in actual value to four decimal points. As shown later, the discharge current trace is highly noisy due to the use of discrete particles in the hybrid algorithm, which may cause the non-zero W_1 metric values. Note that for this physical model, the W_1 metric can be considered to have a unit of Amps.

Because HPHall is more complex that the Lorenz system, the other outputs of the system such as the thrust and efficiency can be compared to obtain a more holistic understanding of the differences between solutions. For the dynamic frame method, the best parameter yields 1.25% less thrust, 1.3% less specific impulse, and 0.77% less efficiency than the reference solution. The

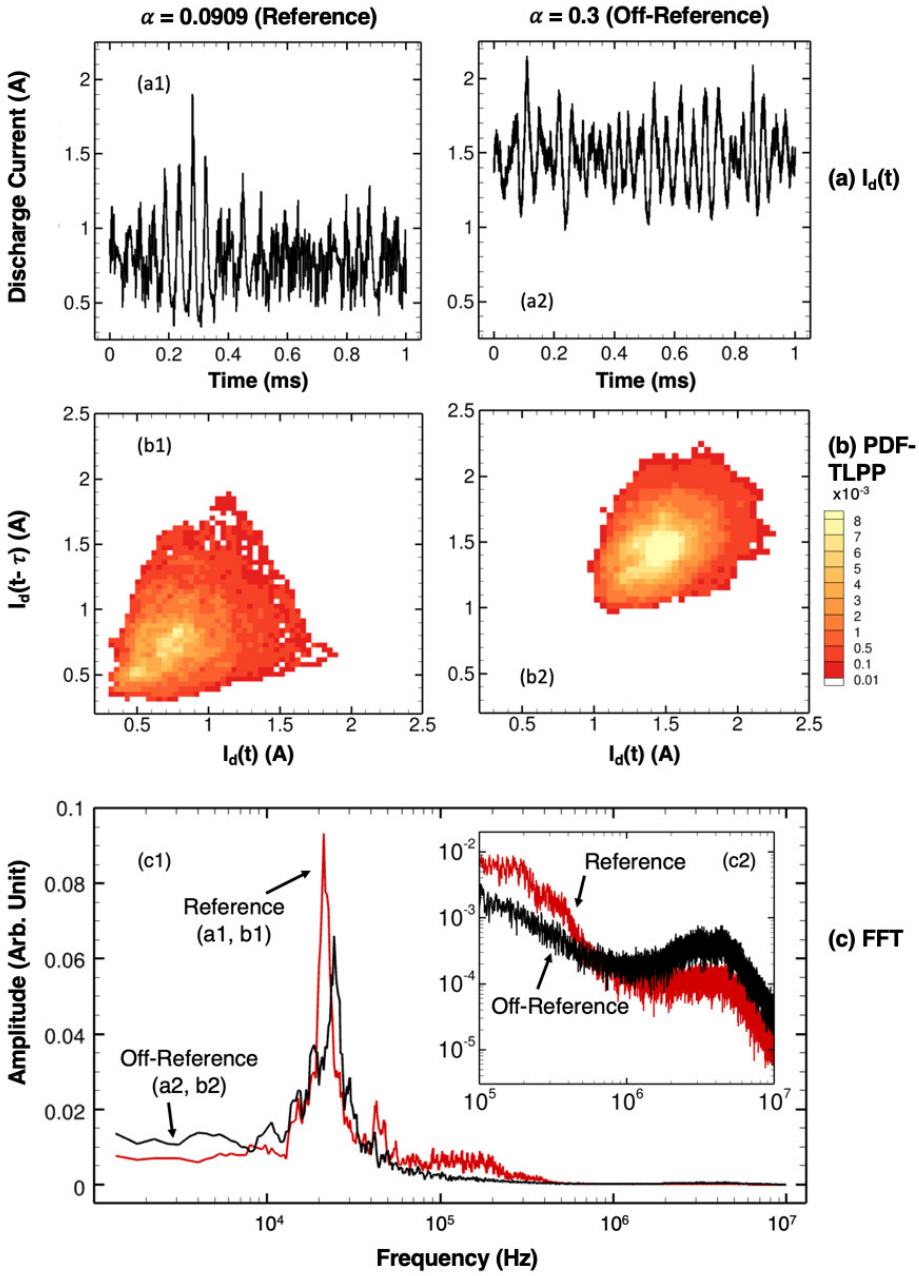

Figure 5.2: A comparison between reference solution results and off-reference solution results. (a1) The discharge current signal from the reference solution. (b1) The discharge current signal from an off-reference solution. (a2) The PDF-TLPP of the reference solution. (b2) The PDF-TLPP of the off-reference solution. Note that unlike the other figures in this work, the color gradient has been reversed so that the lighter yellow signifies the more densely populated regions while the red signifies the less populated regions. (c1) A comparison of the power density spectra for both the reference and off-reference solution with (c2) an enlarged image of the latter half of the plot shown with a log-log scale.

Table 5.1: Optimization of α for a HET discharge plasma using HPHall.

	Dynamic	Static	
		Run 1	Run 2
α range	$0.05 - 0.2$	$0.08 - 0.1$	$0.085 - 0.095$
Optimal α (ε_α)	0.089 (-2.1%)	0.0907 (-0.22%)	0.0909 ($< 0.01\%$)
W_1 metric (A)	**0.029**	**0.0203**	**0.0201**
Current, I_d (ε_I)	0.777 A (-1.74%)	0.792 A ($+0.15\%$)	0.791 A ($+0.05\%$)
Thrust, T (ε_T)	12.4 mN (-1.25%)	12.6 mN ($+0.21\%$)	12.6 mN ($+0.15\%$)
Isp (ε_{Isp})	1476.71 s (-1.3%)	1498.61 s ($+0.21\%$)	1497.74 s ($+0.15\%$)
η (ε_η)	46.39% (-0.77%)	46.88% ($+0.26\%$)	46.87% ($+0.25\%$)

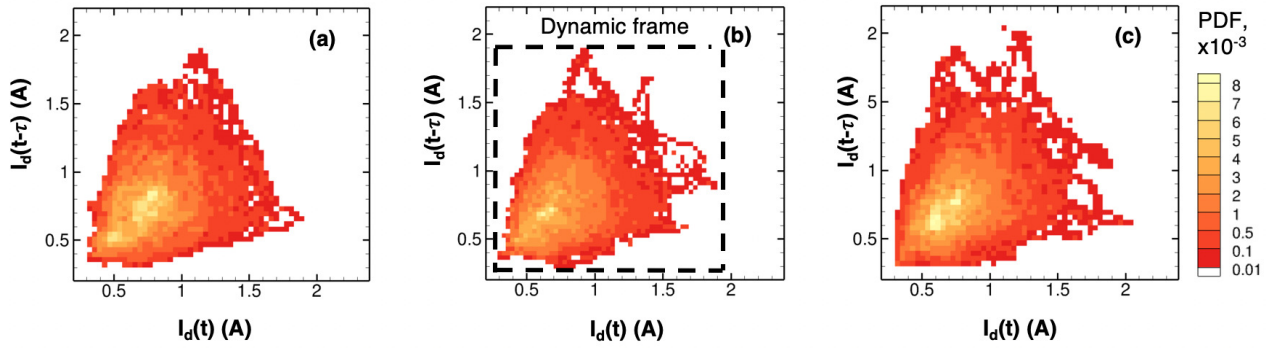

Figure 5.3: The PDF-TLPP for each run case of HPHall. (a) The reference solution. (b) The best dynamic frame method. (c) The best static frame method. Note that the same frame size is chosen for illustration, but the dynamic case (b) shows the actual bounds of the dynamic frame used during the optimization.

first static method test yields a 0.21% greater specific impulse, 0.21% greater thrust, and 0.26% greater efficiency than the reference solution while the second yield even smaller discrepancies. The agreement of the thrust performance other than the discharge current (the only quantity used for calibration), provides strong indication that the overall system state-space behavior, captured by the binned PDF-TLPP, is becoming identical to the reference solution, even if the direct (non-embedded) output signal trajectories remain distinct. While the second static frame case (Run 2) shows a fractional improvement to the first static case (Run 1) when comparing observable quantities, the discrepancy between the α trial and reference values is less than 0.01%, which shows that the data-driven model is successful in finding the *optimal* input condition, the Bohm mobility coefficient. In spite of the optimizer reaching W_1 metric distances smaller than the average current for this plasma discharge model, the W_1 metric values for the optimized solutions is non-zero ($\sim$ 0.02 A). As shown in Fig. 3.7, it is difficult to achieve a solution below a 10^{-2} convergence [91]. This could necessitate a different optimization algorithm being employed once the optimizer has nearly converged to the reference solution to further refine the optimal solution.

Figure 5.3 details the reference solution, dynamic frame, and static frame discharge current plots and resulting PDF-TLPPs, respectively. It must be noted that all of the plots are generated without the first 10% of the discharge current data as there exists an initial spike in the current that eventually settles to a quasi-steady state. Each of the three discharge current plots can be seen to oscillate around approximately the same average value with fluctuating amplitudes below ranging from approximately 0.25 A to 2 A. The densest contour exists near the bottom left corner of the plot ($I_d \sim 0.8$ A) with a lighter distribution emanating away from the corner. Although differences can be seen in the tail of the distributions, such small PDFs only result in a smaller contribution to the overall W_1 metric.

The same numerical errors that are present in the Lorenz system, namely, the numerical errors (e.g., discretization, truncation, round-off) due to the selected numerical scheme, the numerical inaccuracies of the PDF-TLPPs (frame misalignment), and the errors associated when calculating the W_1 metric based on discrete PDF-TLPPs, are still present in the Hall thruster discharge

model. The added difficulty of the plasma discharge model stems from the particle noise in the model, prominent in any particle-based simulation, and the large number of degrees of freedom, i.e., number of macroparticles, in the system interacting with and affecting the numerical uncertainties previously discussed. Despite these uncertainties, the PDF-TLPP and W_1 metric proved a beneficial optimization method for model calibration, showing promising step-forward in utilizing such data-driven model to investigate nonlinear dynamics of physical phenomena. One of the first steps for this future work will be to perform a study of the time lag, τ, sensitivity and how many delay embeddings are required for this Hall thruster model and other physical systems [64].

6. AN ONLINE VALIDATION METHOD

The work in this thesis was performed in conjunction with an *online* validation method for Hall effect thruster research. For this work, the same zero-dimensional bulk plasma model from the previous section is used as a proof of concept. The main premise of this online validation method is that if given access to the time history of some trusted output signal, one should be able to find the best estimate for the time history of the desired input condition. This is often called the unknown input estimation problem in dynamical systems theory [92, 93]. The simplified premise of this technique is to estimate the unknown input using the observations of all or part of the state vector of the system [51]. This is borne from the notion that if parameters of the process vary in time, the most effective way to identify the parameter is to track the parameter in real-time, also known as *real-time identification* [94]. This model will use an extended Kalman filter method for the online modeling.

For this model, assume that the state vector $\mathbf{z(t)}$ consists of both the ion and neutral number densities. Then, the dynamical system of interest can be written as

$$\dot{\mathbf{z}} = \mathbf{g(z)} + \mathbf{h(z)}\xi(\mathbf{t}), \tag{6.1}$$

where $\mathbf{g}$ is the first term of the equation and $\mathbf{h}$ contains the terms multiplied with ξ on the right hand side of equations 4.1a and 4.1b. If we consider $\xi(t)$ as the unknown input to this system, written as $\hat{\xi}(t)$, we can use part of all of the observations of $z(t)$ to estimate it. This can be performed in real-time during the simulation, which is why it often gets employed in control systems processes. Note that for a physical problem only a function of the state or the partial state may be available for real-time estimation and the available state may also be corrupted by noise. Noise can be included as

$$\bar{y}_k = \mathbf{h}(\mathbf{z}(\mathbf{t_k})) + \nu_{\mathbf{k}}, \tag{6.2}$$

where the noise, ν, is used to form the measurement $\bar{y}$ for a discrete time k. This general system representation is employed in an extended Kalman filter (EKF) to reconstruct the temperature profile of the zero-dimensional plasma model. The extended Kalman filter has been a known and proven technique for the last number of decades [94]. Its application to system identification was first proposed by Kopp and Orford [95] and has been applied across several areas of research and industry including nuclear reactors [96] and heat box dynamics [97]. The EKF equations of interest are as follows:

$$\mathbf{x(t)} = \left[\mathbf{z^T}\xi(\mathbf{t})\right]^{\mathbf{T}}, \tag{6.3}$$

where the unknown input $\xi(t)$ is augmented to the original state, $\mathbf{z}$. The extended Kalman filter predicates on a few assumptions regarding the nature of the signal. Notably, these assumptions pertain to the smoothness of the signal in question. If the system is assumed piecewise-constant between two measurement instances, the unknown input can be defined as w, a Brownian motion process defined by the power spectral density $w(t)\ N(0, q)$. The model is propagated as

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{t}) + \mathbf{Gw}, \tag{6.4}$$

where $G = [0\ \ 0\ \ 1]^T$.

Artificial noise with a zero-mean variation is used to generate the 'experimental' measurements in the data using Eq. 6.2, and shown in Fig. 6.1. Note that for visualization purposes, the model used in Fig. 6.1 is a normalized, non-dimensional model. This figure is provided to show how the user-defined estimation is applied in the model, in such a way that emulates comparing computational models and noisy experimental data.

The Kalman filter solution is intentionally initialized with a different, non-ideal $\hat{x}(t_0)$ than the reference solution. The optimal filter gain is computed as

$$K_k = P_k^- H_k^T(\hat{x}_k)\left[H_k(\hat{x}_k)P_k^- H_k^T(\hat{x}_k) + R_k\right]^{-1}, \tag{6.5}$$

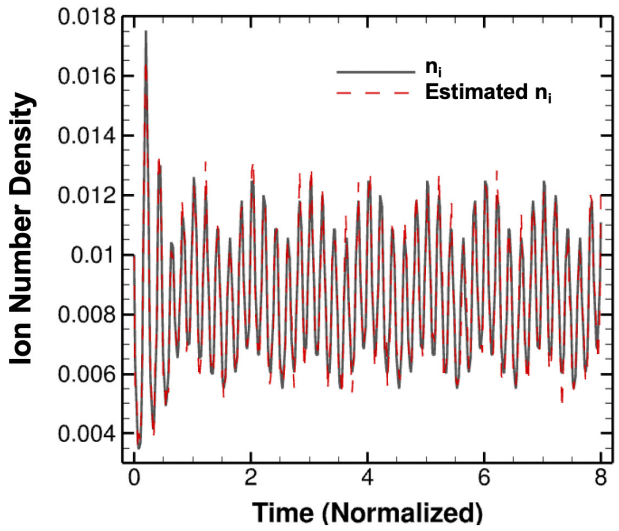

Figure 6.1: An example of the true ion number density plot and the resulting estimated ion number density plot using a 10% zero-mean measurement error.

where R_k is the measurement error and $H_k(\hat{x}_k) = [1 \;\; 0 \;\; 0]$. The state propagation is performed by

$$\dot{\hat{\mathbf{x}}} = \mathbf{f}(\hat{\mathbf{x}}, \mathbf{t}), \tag{6.6}$$

$$\dot{P}(t) = F(\hat{\mathbf{x}}, t)P(t) + P(t)F^T(\hat{\mathbf{x}}, t) + GG^T q, \tag{6.7}$$

where $F(\hat{x}, t) = \left[\frac{\delta f}{\delta x}\right]_{\hat{x}}$ is the Jacobian matrix. Further detail regarding the development of this method is given in Ref. [51].

Using a piecewise constant assumption in the EKF for the smoothness of the data is a rather general assumption to make, meaning that any further restrictions to the smoothness of the ionization rate coefficient will only improve the results shown in this section. The effort presented here is to show that even worst-case, the EKF performs admirably to reconstruct the unknown input signal trace. The model provided as an example in this work is a dimensional oscillatory plasma model that will in the future become a breathing-mode oscillation seen in Hall effect thrusters [23]. These oscillations have been seen in both computational models and experimental results. Using the same

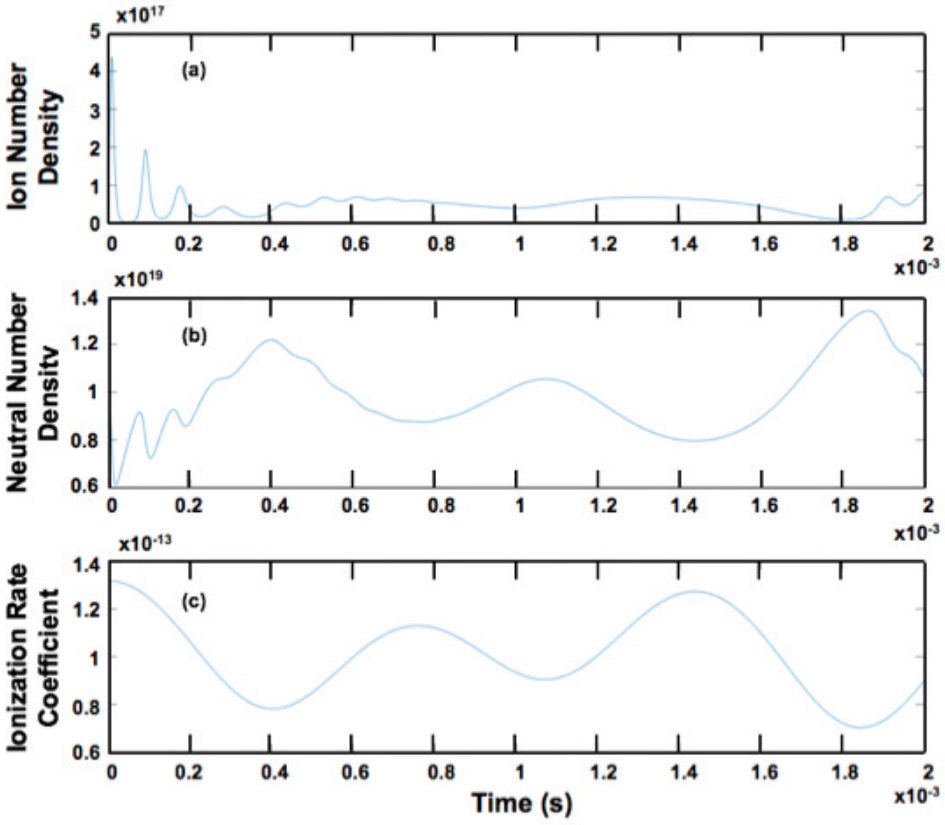

Figure 6.2: The original, reference, signal for the (a) ion number density, (b) neutral number density, and (c) ionization rate coefficient. Note that the temperature input causes the ionization rate coefficient through Eq. 4.2.

SPT-100 thruster geometry and constants from Chap. IV, but applying a different frequency to the electron temperature oscillations, the following reference solution is generated.

The EKF model is allowed to use the state of the ion number density to reconstruct the electron temperature signal seen in Fig. 6.2. The estimated solution and associated error plots are given below.

The extended Kalman filter is shown to reach the original oscillation frequency and amplitude of the electron temperature equation by the completion of the real-time simulation, seen in Fig. 6.3. The length of time required to reach the exact solution is dependent on the initial covariance values chosen for each parameter as well as the process noise term, which dictates how flexible

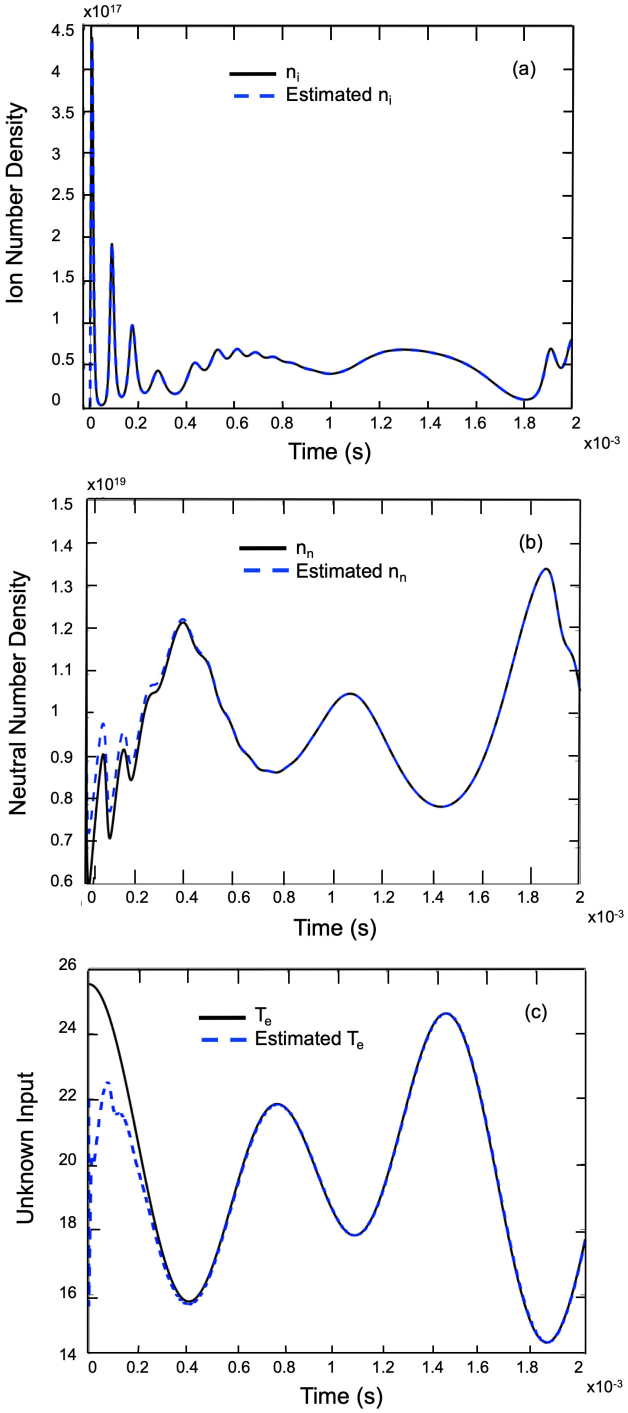

Figure 6.3: The time-dependent estimated values for the (a) ion number density, (b) neutral number density, and (c) temperature input.

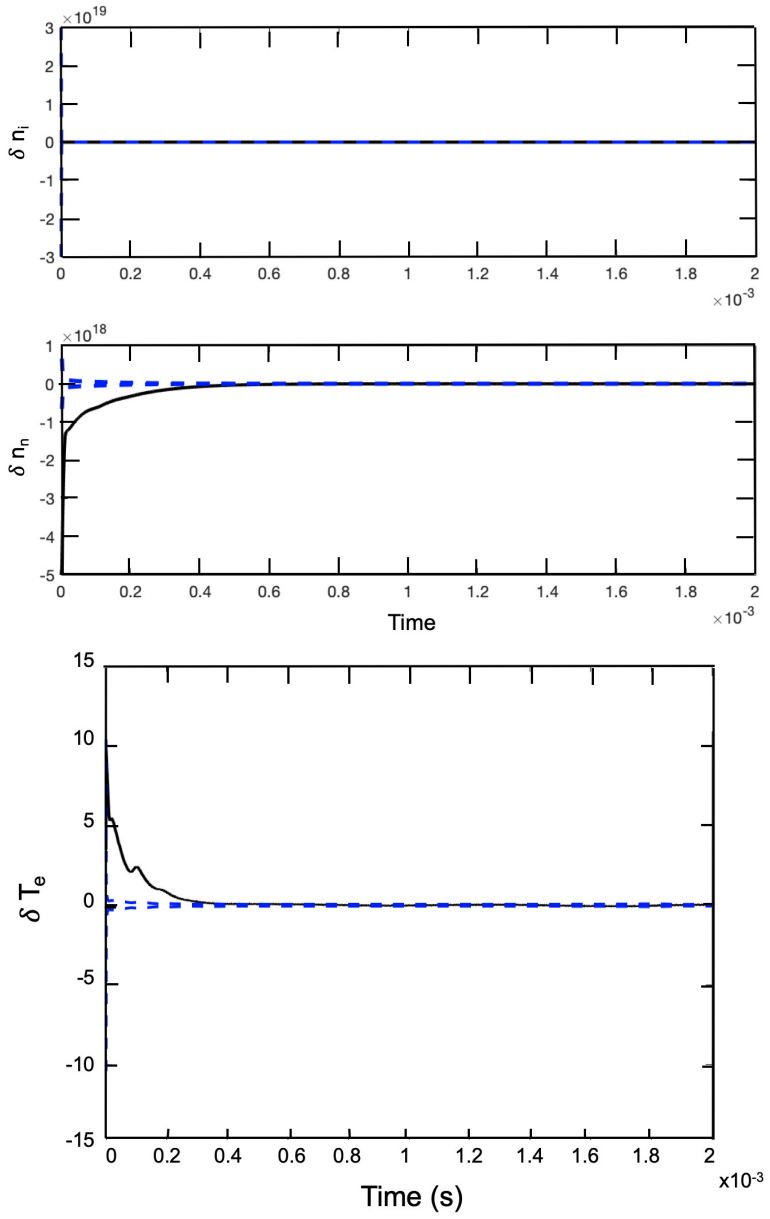

Figure 6.4: The time-dependent error values for the (a) ion number density and neutral number density, and (b) temperature input. The blue dashed lines represent the error bounds while the black solid line is the error itself.

the covariance parameters are to being changed. An example of the errors corresponding to the estimation process incurred by the estimator is output as part of this model's functionality. The true errors are subtracted from the estimates and plotted for the states and unknown inputs as demonstrated in Fig. 6.4. The error covariance is plotted around the error signal to demonstrate

that the EKF captures the errors in the physics without breaking the measurement errors incurred in the model. For this oscillatory mode of plasma behavior, the covariance process can actually be seen to 'breathe' with the errors incurred in the estimate. Though not visible in this non-normalized case, see Ref. [51] for a normalized model that clearly shows such behavior.

These highly preliminary results demonstrate how the extended Kalman filter can be used to estimate the time-dependent signal of plasma properties in Hall effect thrusters. A more in-depth study regarding this work, its abilities and limitations, and its application to Hall thruster models is in progress to become a journal article in the near future.

7. SUMMARY AND CONCLUSIONS

The data-driven model calibration method presented in this thesis is shown to recover the reference solution parameters to high levels of accuracy for a variety of nonlinear, dynamic, chaotic computational models. The initial testing performed with the Lorenz system demonstrates that the frame dependence for a binned solution can reduce the lowest achievable Wasserstein distance by 50%. This study revealed an effective strategy whereby a dynamic frame is used to reduce the initial size of the parameter phase space and a static frame is then used to further optimize a small range of parameter values. The uniqueness of the Lorenz system solution is shown through the comparisons of various sampling lengths and a sensitivity analysis. The sensitivity of the input parameters to infinitesimal changes up to double-precision perturbations was investigated.

The zero-dimensional plasma model was used to study the convergence properties of a plasma physics model using the proposed calibration method. The convergence of the discretization of the probability distribution function binning, subsets of data being used, and how long of a simulation is necessary to negate differences in run time were all quantified to better understand how differences between computational and experimental data may occur.

The zero-dimensional plasma model was also used with a variety of discrete and continuous solution forms to understand how the calibration method handles a difference in solution forms. While the results yield noticeably higher W_1 metric values than other solutions presented in this thesis, the calibration method is at least able to understand a lack of need for all of the equation modes when using a continuous solution to describe a discrete solution, and the location of the more significant frequencies by a discrete equation form when attempting to replicate a continuous system. This demonstrates the solution's ability to identify the significant dynamics is a system.

The model calibration method was then used with a Hall effect thruster model to calibrate the electron mobility parameter used in the model. The calibration method found a solution within 10^{-4} of the reference value. Output signals not used as part of the calibration method, such as the thrust and specific impulse of the Hall-effect thruster model, were accurate to within 1% of

the reference solution values. This implies that the calibration method is capable of determining a form of causality between input parameters and output signals such that an optimal solution can be discovered.

Lastly, a more active model that performs calibration real-time during the simulation was presented. This method is shown as capable of finding the true solution time-dependent trace of the zero-dimension plasma model, though more work to this end is to be performed in the future.

7.1 Further Study

As mentioned in the introduction, this calibration method is only the basest of models that could be developed for data-driven modeling of Hall effect thruster studies. The static, or steady-state, solution generated by this method is useful, but limited in its abilities. The assumption of a steady-state solution is likely an oversimplification of the steady-state operation mode of a Hall effect thruster, which still experiences dynamic oscillations in that operating mode. Instead, the development of a dynamic (in time and space) solution must be determined to account for all modes of a Hall thruster that require different operating conditions. The discoveries made through this work will assist in guiding the development of a dynamic calibration method, but certainly do not constitute as a complete solution. The simple lack of data-driven modeling techniques applied specifically to plasma propulsion leave a broad field of potential new insights to computational modeling.

REFERENCES

[1] NASA, Solar System Exploration: Dawn, *https://solarsystem.nasa.gov/missions/dawn/overview/*, 2019.

[2] A.D. Gallimore, A.F. Thurnau, "The Physics of Spacecraft Hall-Effect Thrusters", *Plasmadynamics and Electric Propulsion Laboratory*, University of Michigan, Department of Aerospace Engineering, 2008.

[3] R. Hofer, L.K. Johnson, D.M. Goebel, and R.E. Wirz, "Effects of Internally Mounted Cathods on Hall Thruster Properties," *IEEE Transactions on Plasma Science* Vol. 36, No. 5-20004-2014, 2008.

[4] I. Katz, V.H. Chaplin, and A.L. Ortega, "Particle-in-cell Simulations of Hall thruster acceleration and near plume regions," *Physics of Plasmas* Vol. 25, 123504, 2018.

[5] K. Hara, "An overview of discharge plasma modeling for Hall effect thrusters", *PSST* Vol. 28, No. 044001, 2019.

[6] J.M. Fife, "Hybrid-PIC modeling and electrostatic probe survey of Hall thrusters", Ph.D. Dissertation, Dept. of Aeronautics and Astronautics, Massachusetts Institute of Technology, 1998.

[7] W.A. Hargus and M.R. Nakles, "Hall Effect Thruster Ground Testing Challenges," AFRL Technical Paper, *American Testing Society 2009*, Huntington Beach, CA, 2009.

[8] T. Lafleur, S.D. Baalrud, and P. Chabert, "Theory for Anomalous Electron Transport in Hall Effect Thrusters. II. Kinetic Model", *Physics of Plasmas* Vol. 23, No. 5-053503, 2016.

[9] S. Cho, H. Watanabe, K. Kutoba, S. Iihara, K. Fuchigami, K. Uematsu, and I. Funaki, "Study of Electron Transport in a Hall Thruster by Axial-Radial Fully Kinetic Particle-In-Cell Simulation," *Physics of Plasmas*, Vol. 22, No. 103523, 2015.

[10] M.D. Campanell, A.V. Khrabrov, and I.D. Kaganovich, "General Cause of Sheath Instability Identified for Low Collisionality Plasmas in Devices with Secondary Electron Emission", *Phys. Rev. Lett.*, Vol. 108, No. 23-235001, 2012.

[11] F. Taccogna and P. Minelli, "Three-Dimensional Fully Kinetic Particle-In-Cell Model of Hall-Effect Thruster", *International Electric Propulsion Conference 2011*, No. 088, 2011.

[12] M. Hirakawa and Y. Arakawa. "Particle simulation of plasma phenomena in Hall thrusters", *Proc. 24th International Electric Propulsion Conference*, Moscow, 1995, pp. 95-164.

[13] J.M. Fife, "Two-dimensional hybrid particle-in-cell modeling of Hall thrusters", *Master's of Science Thesis*, MIT Dept. of Aerospace Engineering, 1995.

[14] E. Ahedo, P. Martinez-Cerezo, and M. Martinez-Sanchez, "One-dimensional model of the plasma flow in a Hall thruster", *Physics of Plasmas* Vol. 8, 2001.

[15] A.I. Smolyakov, O. Chapurin, W. Rias, O. Koshkarov, I. Romadanov, T. Tang, M. Umansky, Y. Raitses, I.D. Kaganovich, and V.P. Lakhin, "Fluid theory and simulations of instabilities, turbulent transport and coherent structures in partially-magnetized plasmas of ExB discharges", *Plasma Physics and controlled Fusion* Vol. 59, No. 014041, 2017.

[16] S. Barral, K. Makowski, and Z. Peradzynski, "Wall material effects in stationary plasma thrusters. II. Near-wall and in-wall conductivity", *Physics of Plasmas* Vol. 10, 2003.

[17] S. Barral, E. Ahedo, "Low-frequency model of breathing oscillations in Hall discharges", *Physical Review E* Vol. 79, No. 4-046401, 2009.

[18] I.G. Mikellides and I. Katz, "Numerical Simulations of Hall-effect plasma accelerators on a magnetic-field-aligned mesh", *Physical Review E* Vol. 86, No. 046703, 2012.

[19] R.R. Hofer and I. Katz, I.G. Mikellides, D.M. Goebel, K.K. Jameson, R.M. Sullivan, and L.K. Johnson, "Efficacy of Electron Mobility Models in Hybrid-PIC Hall Thruster Simulations", 44th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit, Hartford, CT, July 2008.

[20] B. Jorns , "Predictive, data-driven model for the anomalous electron collision frequency in a Hall effect thruster," *Plasma Sources Sci. Technol.* Vol. 27, No. 104007, 2018.

[21] K. Hara and K.M. Hanquist, "Test cases for grid-based direct kinetic modeling of plasma flows", *Plasma Sources Science and Technology*, Vol. 27, No. 6-065004, 2018.

[22] G. J. M. Hagelaar, J. Bareilles, L. Garrigues, and J. P. Boeuf, "Role of anomalous electron transport in a stationary plasma thruster simulation", *J. Appl. Phys.*, Vol. 93, No. 67, 2003.

[23] J.P. Bouef and L. Garrigues, "low frequency oscillations in a stationary plasma thruster", *Journal of Applied Physics* Vol. 84, 1998.

[24] A.I. Morozov, A.P. Shubin, "Kinetics of electrons in the by-wall conductivity regime I.", *Journal of Plasma Physics* Vol. 16, No. 24-16082671, 1984.

[25] V. Latocha, L. Garrigues, P. Degond, and J.P. Bouef, "Numerical simulation of electron transport in the channel region of a stationary plasma thruster", *Plasma Sources Science and Technology* Vol. 11, 2002.

[26] J.D. Adam, A. Heron, and G. Laval, "Study of stationary plasma thrusters using two-dimensional fully kinetic simulations", *Physics of Plasmas* Vol. 11, 2004.

[27] J.W. Koo and I.D. Boyd, "Modeling of anomalous electron mobility in Hall thrusters", *Physics of Plasmas* Vol. 13, No. 3-033501, 2006.

[28] D. Solomatine, L.M. See, and R.J. Abrahart, "Data-driven modeling: Concepts, approaches, and experiences", *Practical Hydroinformatics: Computational Intelligence and Technological Developments in Water Applications*, Springer, Berlin, 2008, pp.17-30.

[29] M.S. Parsons, "Interpretation of machine-learning-based disruption models for plasma control", *Plasma Physics and Controlled Fusion*, Vol. 59, No. 085001, 2017.

[30] W.D. Anderson, D. DeCicco, J.S. Schwaber, and R. Vadigepalli, "A data-driven modeling approach to identify disease-specific multi-organ networks driving physiological dysregulation", *PLoS Comput Biol*, Vol. 13, No. 1005627, 2017.

[31] H. Orouji, O. B. Haddad, E. Fallah-Mehdipour, and M. A. Mariño, "Modeling water quality parameters using data-driven models", *Journal of Environmental Engineering*, Vol. 139, 2013, pp. 947-957.

[32] B. Liu, M. Plumlee, and E. Byon, "Data-driven parameter calibration in wake models", 2018 Wind Energy Symposium, AIAA SciTech Forum, No. 210029, 2018.

[33] J. Bongard and H. Lipson, Proceedings of the National Academy of Sciences, Vol. 104, No. 24, 2007, pp. 9943-48.

[34] P.J. Schmid, "Dynamic mode decomposition of numerical and experimental data", *Jour. Fluid Mechanics*, Vol. 656, 2010, pp. 5-28.

[35] J.N. Kutz, *Data-Driven Modeling and Scientific Computation: Methods for Complex Systems & Big Data*, 1st ed., Oxford University Press, Oxford, 2013.

[36] T. Hastie, R. Tibishirani, and J. Friedman, *The Elements of Statistical Learning*, 2nd ed., Springer, 2009.

[37] R.G. Baraniuk, "Compressive Sensing [Lecture Notes]", IEEE Signal Processing Magazine, Vol. 24, No. 4, 2007, pp. 118-121.

[38] B. Bhattacharya, A.H. Lobbrecht, and D.P. Solomatine, "Neural Networks and Reinforcement Learning in Control of Water Systems", *Jour. Water Resources Planning and Management* Vol. 129, No. 6, 2003, pp. 458.

[39] O. Nelles, *Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models*, Springer, New York, 2001.

[40] J.N. Kutz, S.L. Brunton, B.W. Brunton, and J.L. Proctor, *Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems*, SIAM, 2016.

[41] M. Raissi and G.E. Karniadakis, "Hidden physics models: Machine learning of nonlinear partial differential equations", *Journal of Computational Physics* Vol. 357, 2018, pp. 125-141.

[42] J. Bongard and H. Lipson, "Automated reverse engineering of nonlinear dynamical systems", *PNAS* Vol. 104, No. 24, 2007, pp. 9943-9948.

[43] E. A. Nadarya, "On estimating regression", *Theory Pb. Appl.*, 1964.

[44] Q. He, L. Wang, and B. Liu, "Parameter estimation for chaotic systems by particle swarm optimization", *Chaos, solitons, & Fractals* Vol. 32, No. 2, 2007, pp. 654-661.

[45] F.A. Guerra and L.S. Coelho, "Multi-step ahead nonlinear identification of Lorenz's chaotic system using radial basis neural network with learning by clustering and particle swarm optimization", *Chaos, Solitons, & Fractals* Vol. 35, No. 5, 2008, pp. 967-979.

[46] L.A. Aguirre and C. Letellier, "Modeling Nonlinear Dynamics and Chaos: A Review", *Mathematical Problems in Engineering*, No. 238960, 2009.

[47] B. Lusch, J.N. Kutz, and S.L. Brunton, "Deep learning for universal linear embeddings of nonlinear dynamics", *Nature Communications* Vol. 9, 2018.

[48] E. Kaiser, J.N. Kutz, S.L. Brunton, "Sparse identification of nonlinear dynamics for model predictive control in the low-data limit", *Proceedings of the Royal Society A: Mathematical, Physical, and Engineering Sciences*, 2018.

[49] S.H. Rudy, S.L. Brunton, J.L. Proctor, and J.N. Kutz, "Data-driven discovery of partial differential equations", *Science Advances* Vol. 3, No. 4-1602614, 2017.

[50] J.L. Proctor, S.L. Brunton, and J.N. Kutz, "Dynamic mode decomposition with control", *SIAM Journal on Applied Dynamical Systems* Vol. 15, No. 1-1013857, 2016, pp.142-161.

[51] M. Majji, C. Greve, and K. Hara, "Unknown input and state estimation in plasma dynamics for data-driven modeling applications", *AIAA Joint Propulsion Conference & Exhibition Presentation*, August 2019.

[52] P.D. Lax and R.D. Richtmyer, "Survey of the stability of linear finite difference equations" *Communications on Pure and Applied Mathematics* Vol. 9, 1956.

[53] S. Chiocchini, T. Pagliaroli, R. Camussi, and E. Giacomazzi, "Chaotic and Linear Statistics Analysis in Thermoacoustic Instability Detection", *Journal of Propulsion and Power* Vol. 34, No. 15, 2018.

[54] G.E.P. Box, G.M. Jenkins, and G.C. Reinsel, *Time Series Analysis: Forecasting and Control* (Prentice Hall 1994).

[55] L. Ljung, *System Identification: Theory for the User*, Prentice Hall, 1999.

[56] T.K. Huang and J. Schneider, "Learning linear dynamical systems without sequence information", *Proceedings of the 26th International Conference on Machine Learning*, 2009, pp. 425-432.

[57] S.L. Brunton, J.L. Proctor, and J.N. Kutz, "Discovering governing equations from data by sparse identification of nonlinear dynamical systems", *Proceedings of the National Academy of Sciences* Vol. 113, 2016.

[58] G.J. Bowden, G. Dandy, and H. Maier, "Input Determination for Neural Network Models in Water Resources Applications: 2. Background and Methodology", *Journal of Hydrology* Vol. 301, No. 75, 2005, pp. 93-107.

[59] I. Daubechies, "The wavelet transform, time-frequency localization and signal analysis", *IEEE Trans. Inf. Theory* Vol. 36, 1990.

[60] G. Sugihara, R. May, H. Ye, C. Hsieh, E. Deyle, M. Fogarty, and S. Munch, "Detecting Causality in Complex Ecosystems," *Science*, Vol. 338, 2012, pp. 496-500.

[61] A. Basharat and M. Shah, "Time Series Prediction by Chaotic Modeling of Nonlinear Dynamical Systems", University of Central Florida, 2009.

[62] F. Takens, "Detecting Strange Attractors in Turbulence," *Dynamical Systems and Turbulence*, Warwick, 1981, pp. 366-381.

[63] L. Cao, A. Mees, and K. Judd, "Dynamics from multivariate time series," *Physica D: Nonlinear Phenomena*, Vol. 121, No. 1-2, 1998, pp. 75-88.

[64] R. Martin, J. Koo, D. Eckhardt, "Impact of Embedding View on Cross Mapping Convergence", arXiv, Cornell University, Ithaca, NY, arXiv:1903.03069, 2019.

[65] S.H. Cha, "Comprehensive Survey on Distance/Similarity Measures between Probability Density Functions", *International Journal of Mathematical Models and Methods in Applied Sciences* Vol. 1, No. 4, 2007, pp. 300-307.

[66] J.W. Koo, R.S. Martin, D.Q. Eckhardt, "Improved calibration techniques for hybrid PIC HET codes", *Air Force Research Laboratory*, 2018.

[67] S. Bekta and Y. Sisman, "The comparison of L1 and L2-norm minimization methods", *Int. J. Phys. Sci.* Vol. 5, 2010, pp. 1721.

[68] Y. Rubner, C. Tomasi, L. J. Guibas, "The Earth Mover s Distance as a Metric for Image Retrieval", *Int. J. Comp. Vis.* Vol. 40, No. 2, 2000, pp. 99-121.

[69] C. Villani, *Topics in Optimal Transportation*, American Mathematical Society, 2003.

[70] Y. Robin, P. Yiou, and P. Naveau, "Detecting changes in forced climate attractors with Wasserstein distance", *Nonlinear Processes in Geophysics* Vol. 24, 2017, pp. 393-405.

[71] M. Muskulus and S. Verduyn-Lunel, "Wasserstein Distances in the Analysis of Time Series and Dynamical Systems", Technical Report MI-2009-12, 2009.

[72] N. Bonneel, M. Van De Panne, S. Paris, and W. Heidrich, "Displacement interpolation using Lagrangian mass transport", *ACM Transactions on Graphics* Vol. 30, 2011, pp.158.

[73] M. S. Eldred, S. L. Brown, B. M. Adams, D. M. Dunlavy, D. M. Gay, L. P. Swiler, W. E. Hart, J. D. Griffin, P. D. Hough, T. G. Kolda, and M. Martinez-Canales, Sandia Technical Report SAND2014-4633, 2014.

[74] Stellar Science LLC, Galaxy Simulation Builder Users Manual v6.7, 2018.

[75] R. Flamary and N. Courty, POT Python Optimal Transport Library 0.5.1, 2018.

[76] E.N. Lorenz, “Deterministic Nonperiodic Flow,” *Jour. of Atmos. Sci.* Vol. 20, 1963, pp. 130.

[77] M. Sommerfeld and A. Munk, "Inference for empirical Wasserstein distances on finite spaces", *Jour. R. Statist. Soc. B*, Vol. 80, 2018, pp. 219-238.

[78] K.P. Champion, S.L. Brunton, and J.N. Kutz, "Discovery of Nonlinear Multiscale Systems: Sampling Strategies and Embeddings", arXiv:1805.07411, 2018.

[79] A.J. Lotka, *Principles of Physical Biology*. Baltimore: Waverly, 1925.

[80] V. Volterra, "Fluctuations in the abundance of a species considered mathematically." *Nature* Vol. 118, 1926, pp. 558-560.

[81] K. Hara, M.J. Sekerak, I.D. Boyd, and A.D. Gallimore, "Perturbation analysis of ionization oscillations in Hall effect thrusters", *Phys. of Plasmas* Vol. 21, 122103, 2014.

[82] N. Gascon, M. Dudeck, and S. Barral, "Wall material effects in stationary plasma thrusters. II. Near-wall and in-wall conductivity", *Physics of Plasmas* Vol. 10, 2003.

[83] F.I. Parra, E. Ahedo, J.M. Fife, and M. Martínez-Sánchez, "A two-dimensional hybrid model of the Hall thruster discharge", *J. of Appl. Phys.* Vol. 100, No. 023304, 2006.

[84] D. Sydorenko, A. Smolyakov,I. Kaganovich, and Y. Raitses, "Modification of electron velocity distribution in bounded plasmas by secondary electron emission", *IEEE Transactions on Plasma Science* Vol. 34, 2006, pp.856.

[85] A. Héron and J.C. Adam, "Theory for the anomalous electron transport in Hall effect thrusters. II. Kinetic model", *Physics of Plasmas* Vol. 20, No. 8-082313, 2013.

[86] S. Janhunen, A. Smolyakov, O. Chapurin, D. Sydorenko, I. Kaganovich, and Y. Raitses, "Nonlinear structures and anomalous transport in partially magnetized EŒB plasmas", *Physics of Plasmas* Vol. 25, No. 1-011608, 2018.

[87] J.P. Boeuf and L. Garrigues, "EŒB electron drift instability in Hall thrusters: Particle-in-cell simulations vs. theory", *Phys. Plasmas*, Vol. 25, No. 6-061204, 2018.

[88] W.A. Hargus Jr., "Laser-Induced-Fluorescence-Derived Hall Effect Thruster Ion Velocity Distribution Visualization", *IEEE Transactions on Plasma Science* Vol. 39, 2011, pp. 2918-2919.

[89] I. Romadanov, Y. Raitses, A. Diallo, K. Hara, I.D. Kaganovich, and A. Smolyakov, "On limitations of laser-induced fluorescence diagnostics for xenon ion velocity distribution function measurements in Hall thrusters", *Physics of Plasmas* Vol. 25, No. 3-033501, 2018.

[90] B.M. Reid, "The Influence of Neutral Flow Rate in the Operation of Hall Thrusters", *PhD Dissertation*, University of Michigan Department of Aerospace Engineering, 2009.

[91] J.A. Lazzus, M. Rivera, and C.H. López-Caraballo. "Parameter estimation of lorenz chaotic system using a hybrid swarm intelligence algorithm," *Phys. Lett. A*, Vol. 380, 2016, pp. 1164.

[92] K.S. Narendra and A.M. Annaswamy, *Stable Adaptive Systems*, Dover Pulbications, Mineola, NY, 2005.

[93] P. Ioannou and J. Sun, *Robust Adaptive Control*, Dover Publications, Mineola, NY, 1996.

[94] K.J. Astrom and P. Eykhoff, "System identification: A survey", *Automatica*, Vol. 7, No. 2, 1971, pp. 123-162.

[95] R. E. Kopp and R. J. Orford, "Linear regression applied to system identification for adaptive control systems", *AIAA Journal*, 1963, pp. 2300-2306.

[96] L. J. Habegger and R. E. Bailey, "Minimum variance estimation of parameters and states in nuclear power systems", *Proc. fourth IFAC congress*, Warsaw, No. 12.2, 1969.

[97] A. Baggerud and J. G. Balchen, "An adaptive state estimator", *2nd IFAC Symposium: Identification and process parameter estimation*, Prague, No. 10.1, 1970.