REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED *(From - To)*
23 September 2020	Technical Paper with Briefing Charts	08 September 2020 - 30 September 2020

4. TITLE AND SUBTITLE
RIPS 2020 Final Report: An Unstructured Mesh Approach to Nonlinear Noise Reduction (Technical Paper with Briefing Charts)

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
M. DeBrito, J. Botvinick-Greenhouse, A. Kirtland, M. Osborne, C. Johnson, R. Martin, D. Eckhardt

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER
Q2DF

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Research Laboratory (AFMC)
AFRL/RQRS
1 Ara Drive
Edwards AFB, CA 93524-7013

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Research Laboratory (AFMC)
AFRL/RQR
5 Pollux Drive
Edwards AFB, CA 93524-7048

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AFRL-RQ-ED-TR-2020-194

12. DISTRIBUTION/AVAILABILITY STATEMENT
Distribution Statement A: Approved for Public Release; Distribution is Unlimited. PA Clearance Number: 20410; Clearance Date: 21 September 2020.

13. SUPPLEMENTARY NOTES
The U.S. Government is joint author of the work and has the right to use, modify, reproduce, release, perform, display, or disclose the work. Prepared in collaboration with University of Michigan - Ann Arbor, Amherst College, Washington University at St. Louis and University of Scranton. Technical Paper with Briefing Charts. RIPS 2020 AFRL Team Deliverables

14. ABSTRACT
In any type of data acquisition, the event of gathering undesirable noise along with desirable data is inevitable. To denoise signals originating from smooth, chaotic attractors, the Air Force Research Laboratory (AFRL) adapted the time-delay embedding theory of Takens' Theorem (1981) and the causation-detecting method of Convergent Cross Mapping (CCM) to develop a grid-based denoising technique. Given a clean signal from such a dynamical system, AFRL's technique attempts to denoise a corrupted signal observed from the same system. To improve this grid-based method, we implement an unstructured mesh based on triangulations and Voronoi diagrams that better distributes data over mesh cells and improves the accuracy of the reconstructed signal. Our method achieves statistical convergence with known test data and reduces synthetic noise on experimental signals from Hall Effect Thrusters (HETs) with greater success than the grid-based strategy.

15. SUBJECT TERMS
N/A

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Robert Martin
Unclassified	Unclassified	Unclassified	SAR	100	19b. TELEPHONE NUMBER *(Include area code)* N/A

Research in Industrial Projects for Students

Sponsor

Air Force Research Laboratory

Final Report

An Unstructured Mesh Approach to Nonlinear Noise Reduction

Student Members

Marianne DeBrito (Project Manager), *University of Michigan - Ann Arbor*, debrito@umich.edu

Jonah Botvinick-Greenhouse, *Amherst College*

Aaron Kirtland, *Washington University at St. Louis*

Megan Osborne, *University of Scranton*

Academic Mentor

Casey Johnson, casey.johnson@cgu.edu

Sponsoring Mentors

Robert Martin, PhD, robert.martin.101@us.af.mil

Daniel Eckhardt, PhD, daniel.eckhardt.3@us.af.mil

Consultants

Samuel J. Araki, PhD

August 18, 2020

This project was jointly supported by AFRL and NSF Grant ⟨DMS⟩ 1440415.

Abstract

In any type of data acquisition, the event of gathering undesirable noise along with desirable data is inevitable. To denoise signals originating from smooth, chaotic attractors, the Air Force Research Laboratory (AFRL) adapted the time-delay embedding theory of Takens' Theorem (1981) and the causation-detecting method of Convergent Cross Mapping (CCM) to develop a grid-based denoising technique. Given a clean signal from such a dynamical system, AFRL's technique attempts to denoise a corrupted signal observed from the same system. To improve this grid-based method, we implement an unstructured mesh based on triangulations and Voronoi diagrams that better distributes data over mesh cells and improves the accuracy of the reconstructed signal. Our method achieves statistical convergence with known test data and reduces synthetic noise on experimental signals from Hall Effect Thrusters (HETs) with greater success than the grid-based strategy.

Acknowledgments

We would like to first thank our industry mentors, Rob Martin and Dan Eckhardt. Through their support and suggestions about directions to take the problem, we were able to find success in this project and accomplish our goals. Also, we would like to thank our academic mentor, Casey Johnson, who helped us with any and all research and administration problems we ran into over the course of this program. In addition to these mentors, we would like to thank Jun Araki for his time working with us as a consultant. His experience with this project was invaluable as we built our solutions and approaches. Without the support of Susana Serna, Neli Petrosyan, Dave Medina, and the rest of the IPAM team, this project would not have been able to run at all. We are extremely grateful to everyone involved in IPAM, and to our mentors, who were flexible enough to continue their support of this program even through such uncertain times. Additionally, we thank the National Science Foundation, who offered their support of this program through grant No. 1440415. We are proud of the results we have found over the course of this project, and we hope they reflect the commitment and time given by everyone who has supported us.

Contents

REFERENCES

List of Figures

Note: Figures 2.9, 3.5, 4.10, 4.11, 4.12, 4.13, 4.14, 4.23, 4.25, 4.27, 4.28, and 4.29 were produced with a slightly different training and testing method. While it is possible this affected their appearance and the results they show, we have no evidence of this, and we strongly suspect that there would be no apparent differences with the change.

Chapter 1

Introduction

1.1 Sponsors

Research in Industrial Projects for Students (RIPS) is a regular summer program organized by the Institute for Pure and Applied Mathematics (IPAM) at the University of California, Los Angeles (UCLA) in which undergraduate and fresh graduate students participate in sponsored team research projects. With support of the National Science Foundation (NSF) and the program's industry sponsors, IPAM aims to give students mentored experience in mathematical industry.

Through RIPS, this project was sponsored by the Air Force Research Laboratory (AFRL). Motivated by aerospace warfighting applications, AFRL aims to develop and distribute technological advances. In particular, our motivation and mentorship came from AFRL's In-Space Propulsion Branch located at the Edwards Air Force Base.

1.2 Our Proposed Problem

In any type of data acquisition, the possibility of gathering undesirable noise along with desirable data is high, if not certain. The research done on Hall-Effect Thruster (HET) dynamics by the AFRL In-Space Propulsion branch is no exception. Noise can be increased through the use of electronics in acquisition, the presence of unpredictable vibrations in surrounding environments, the type of equipment used, and the like, and can corrupt desirable data enough to prevent researchers from recovering system dynamics.

Recently, AFRL has adapted causality-detection methods of Convergent Cross-Mapping (CCM) in an effort to reconstruct clean versions of noisy HET signals [6], and has partnered with IPAM in past RIPS programs in this pursuit (RIPS 2018). This denoising method is explored in Reference [6] and relies on one clean signal of a smooth chaotic attractor to reconstruct a clean version of a noisy signal from the same dynamical system. The reconstruction method involves placing a mesh over a time-delayed embedding of the clean signal as inspired by CCM and Takens' theorem, and is nearly identical to the method described in Chapter 3 of this report, but utilizes a uniform mesh (in the form of a grid) rather than the newly developed unstructured mesh.

We had reason to believe that an unstructured mesh method would yield higher chances of success in denoising and convergence to an appropriate clean signal mainly because it offers the ability to adapt mesh cell size and shape based on the density gradient of the data. This prevented issues which were realized in the uniform mesh such as empty cells,

density gradient differences across cells, and variance errors which occur with smaller mesh cells.

1.3 Our Team's Approach

Our efforts aimed to extend the work of Reference [6] to unstructured meshes to recover "truly convergent" CCM for noisy data signals. We approached this problem with three major goals:

1. **Development.** We aimed to develop code to reconstruct a clean version of a noisy signal using the same algorithm as AFRL's previous work in Reference [6], but using an unstructured mesh rather than a uniform one. Because of its ability to adapt to data density, we utilized a Voronoi diagram as a mesh and a triangulation as a method of linear interpolation between the average cell values. We opted to code in Python for ease of access, ability to work with large sets of data, and applicability of its available packages and libraries.

2. **Analysis.** After an unstructured mesh algorithm was coded, we analyzed its performance by using known dynamical systems which resemble that of the HET. Following the previous work by AFRL, we used data from the Lorenz attractor, keeping the X signal clean and adding various types of synthetic noise to the Y and Z. The corrupted Y and Z signals were then reconstructed by our algorithm, and the reconstructions were evaluated based on the error between the reconstruction and the original clean Y and Z signals. These errors were quantified using the Pearson Correlation Coefficient (PCC). We then varied parameters within the algorithm such as timestep, number of unstructured mesh cells, dimension of time-delayed embedding, and amount of data used for reconstruction, and plotted the resulting changes in error. Later, we applied our algorithm to the Rossler and Chen attractors to ensure that we could achieve reconstruction success on systems other than the Lorenz attractor. These studies allowed us to compare our algorithm's performance to that of the uniform mesh, to optimize the parameter values for each attractor, and to locate and mitigate causes of outstanding error.

3. **Application.** With an optimized and thoroughly tested reconstruction script complete, we accomplished our ultimate goal to successfully denoise real HET experimental data. We first added synthetic noise to considerably clean HET signals and were able to reconstruct the noiseless signal with low leftover error. To determine the reconstruction success of HET signals with high original noise, we compared reconstructions of the signals with different levels of synthetic noise added to them. Optimization was applied to the HET reconstructions and a satisfactory level of success was achieved.

1.4 Overview of Report

In the following pages, we discuss the background knowledge required to explore the above goals and a detailed explanation of our results and findings, as well as a record of the sources we drew from.

In **Chapter 2: Background**, we describe the mathematical theory which makes our reconstruction method possible, as well as a description of the CCM procedure and AFRL's

past adaptations of it. We also describe our method of error quantification and the Lorenz system which we used as test data. Our initial efforts and thorough analysis were built on this knowledge.

In **Chapter 3: Reconstruction Method**, we detail our entire denoising process, coded in Python. It includes a description of the unstructured mesh construction, the signal reconstruction algorithm, and our additional refinement strategies of interpolation and cell adaptation including linear interpolation and k-means clustering as cell adaptation.

In **Chapter 4: Evaluation of Method**, we offer an explanation of our parameter analysis methods and discussions of our findings with respect to the effect the parameters have on reconstruction error. Parameters discussed include the time-delay size, embedding dimension, ODE solver timestep, and number of cells as it relates to the amount of training data used. In this chapter, we also present signal reconstructions of both known test data and HET experimental data, as well as comparisons between the old uniform mesh method and our new unstructured mesh method. Lastly, we discuss areas of outstanding error and their causes, along with possible strategies to mitigate them.

In **Chapter 5: Summary**, we summarize our findings and our recommended future research questions to follow from this work.

In our appendices, we offer a glossary defining technical terms and notation used, alphabetically ordered (**Appendix A**), and a list of the abbreviations used (**Appendix B**). Terms listed in the glossary are italicized when first introduced in the text. We conclude the report with our **Selected Bibliography Including Cited Works**.

Chapter 2

Background

Given a noisy signal and a clean signal both sampled from the same chaotic attractor dynamical system, our goal was to generate a denoised reconstruction of the noisy signal, with the intent of application to Hall Effect Thruster (HET) signals. Before explaining our reconstruction method, we must first introduce the previously known information on which we based our work.

2.1 The Hall Effect Thruster (HET)

A Hall Effect Thruster (HET) is a type of ion thruster used on in-space robotics and sattelites. It has system dynamic properties resembling that of a smooth chaotic attractor, and has been a topic of interest for researchers since the mid- to late-1900s. Due to high noise levels collected during data acquisition, however, experimental signals are often corrupted in a way which makes understanding the true HET system dynamics difficult.

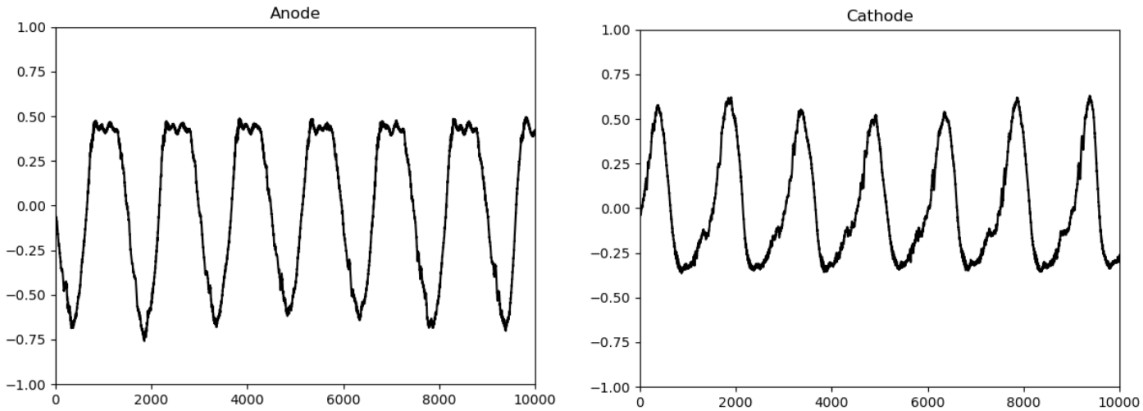


Figure 2.1: Left: Anode Pearson Signal, Right: Cathode Pearson Signal

For the purpose of improving their earlier methods of signal denoising, we were provided with observed signals from HET experiments by the In-Space Propulsion Branch of AFRL in order to test and apply our new method. Specifically, we were given fairly clean signals from the HET in order to make testing efficacy of the algorithm easier. This collected data does have some level of noise, but it is important to note that other collected signals from HET can be significantly noisier. In Figure 2.1, two of these signals are shown, called the

Anode Pearson signal and the Cathode Pearson signal. These names will relate to these specific pieces of data throughout this report.

In Figure 2.2, two additional pieces of data are shown, called the Anode + Cathode signal and the Total Cage signal. These are the primary signals used in creating reconstructions for the HET data.

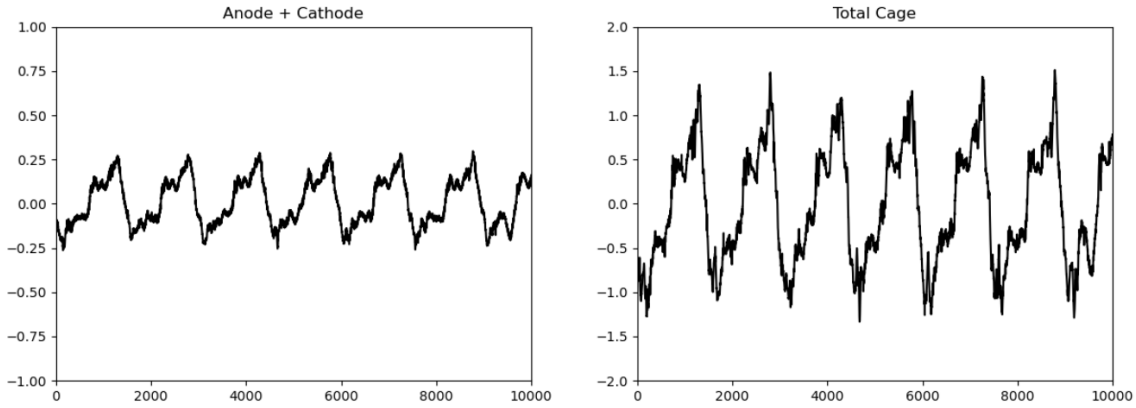


Figure 2.2: Left: Anode + Cathode Signal, Right: Total Cage Signal

Some signals can be collected with relatively little noise, like the Cathode signal, and others are heavily corrupted with noise, like the Ring 1 signal, as Figure 2.3 illustrates. The signals from Figure 2.3 represent two different pieces of data called the Cathode Pearson signal and the Ring 1 signal. Clearly, as these signals illustrate, collected data from the HET system can be highly variable in the amount of noise present.

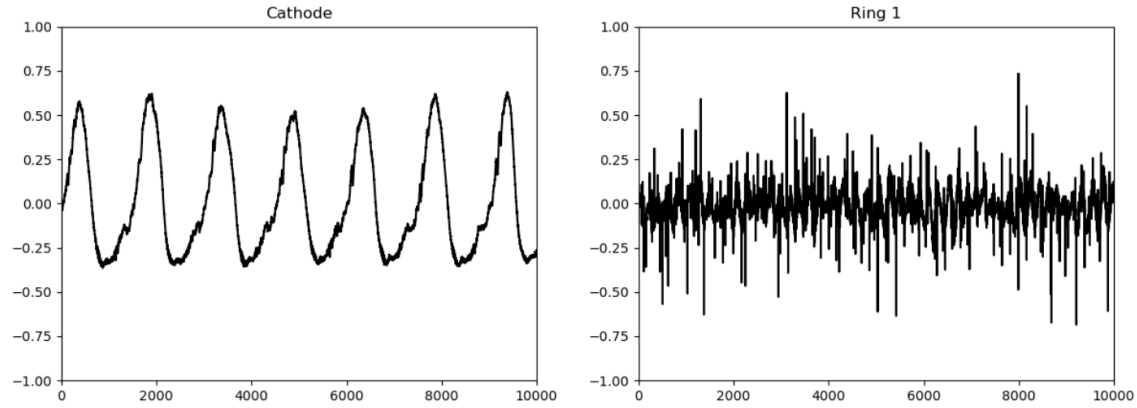


Figure 2.3: Left: Cathode Signal, Right: Ring 1 Signal

2.2 Test Data

In order to develop a denoising algorithm which would accurately recover the desired clean signal, we selected a control set of data which followed similar dynamical charactaristics as those of a HET system. This project primarily utilized the Lorenz attractor system as a test system, though we also investigated the Rossler and Chen attractors for success confirmation. The Lorenz system was specifically chosen because it was the main test system used in past research by AFRL, making it easier to compare current results to past

ones. The Lorenz system was particularly useful in developing our reconstruction algorithm and testing it against known data. The Lorenz system of differential equations is presented in Figure 2.4.

$$\frac{dX}{dt} = \sigma(Y - X)$$
$$\frac{dY}{dt} = X(\rho - Z) - Y$$
$$\frac{dZ}{dt} = XY - \beta Z$$

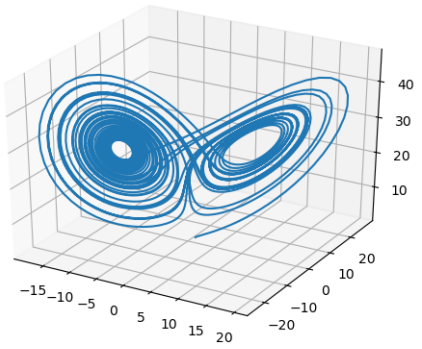


Figure 2.4: Lorenz System

Throughout our simulations, we used the parameters $\sigma = 10, \rho = 30$, and $\beta = 8/3$. Shown in Figure 2.4 is a plot of the Lorenz system with these values. When this manifold is projected onto each axis, it produces a time series. For the Lorenz system, we refer to these time series projects as $X(t), Y(t)$, and $Z(t)$ based on the axis that the data was projected onto. The time series projections of the Lorenz system are displayed in Figures 2.5. These time series are used frequently as we discuss our reconstruction algorithm in later sections.

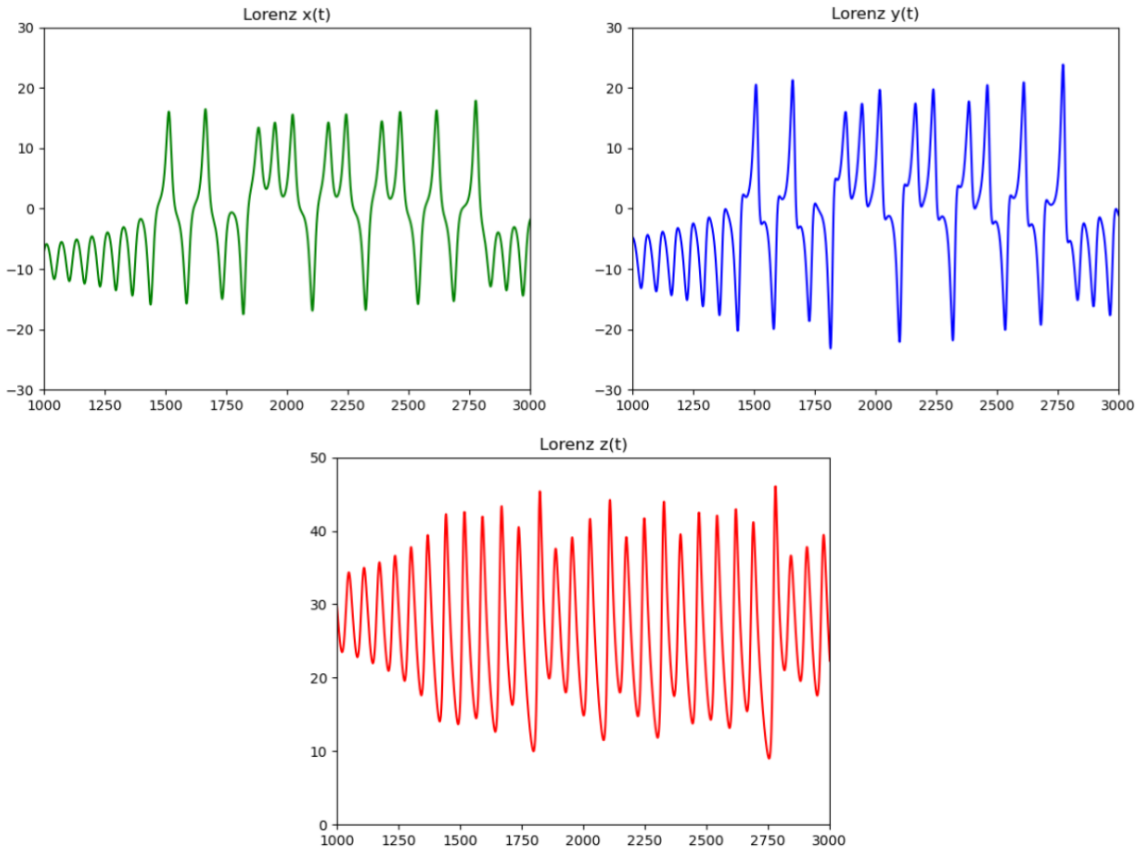


Figure 2.5: X-, Y-, and Z-axis Time Series

2.3 Time-Delay Embedding Theory

The novelty of AFRL's recent denoising explorations is that they attempt to make use of clean signals within a system to recover information about the entire system's dynamics and to reconstruct another signal from the system, which may be noisy. Since we use the clean signal to facilitate this reconstruction, it is crucial to have an understanding of the relationship between the observables of a given dynamical system. In particular, Takens' theorem (1981) is essential in understanding this relationship and is stated as follows.

Theorem 2.1 (Takens' Theorem for Dynamical Systems (simplified)) *For a dynamical system where* $\mathbf{S}$ *is the state of the system, and* $X(t)$ *is a time series projection of the state* $\mathbf{S}(t)$*, there exists some finite dimension* d *and some* τ *such that the manifolds generated by*

$$\mathbf{P}_X(t) := \Big(X(t), X(t+\tau), X(t+2\tau), \ldots, X(t+(d-1)\tau)\Big)$$

and $\mathbf{S}(t)$ *are diffeomorphic.*

We refer to the manifold generated by $\mathbf{P}_X(t)$ as a *time-delay embedding* of the signal $X(t)$, or a *shadow manifold* generated by $X(t)$, and denote it as $\mathcal{M}^X$. Takens' theorem allows for one state variable to recover information about the entire system through the creation of a time-delay embedding, which motivates the use of a clean signal in our noisy signal reconstructions, as we can create a time-delay embedding of the clean signal [6].

The time series projections of the Lorenz system can then be embedded into a specific dimension using time lags of a single signal. Throughout many of our examples, the embedding dimension used is 2; that is, the axes of the X embedding $\mathcal{M}^X$ are $X(t)$ and $X(t+\tau)$ for some positive time lag τ, while the 2D Lorenz system would have axes $X(t)$ and $Y(t)$. Takens' theorem suggests that there exists an optimal least dimension for which the time-delayed embedding adequately represents the entire system, which motivated us to explore the effects of the embedding dimension on our denoising algorithm's success. The similarities between the original system and the time-delayed embedding can even be observed visually, as the shapes of the manifolds in Figure 2.6 are quite similar. Note that the selection of τ is extremely important, and is described in more detail in section 4.1.

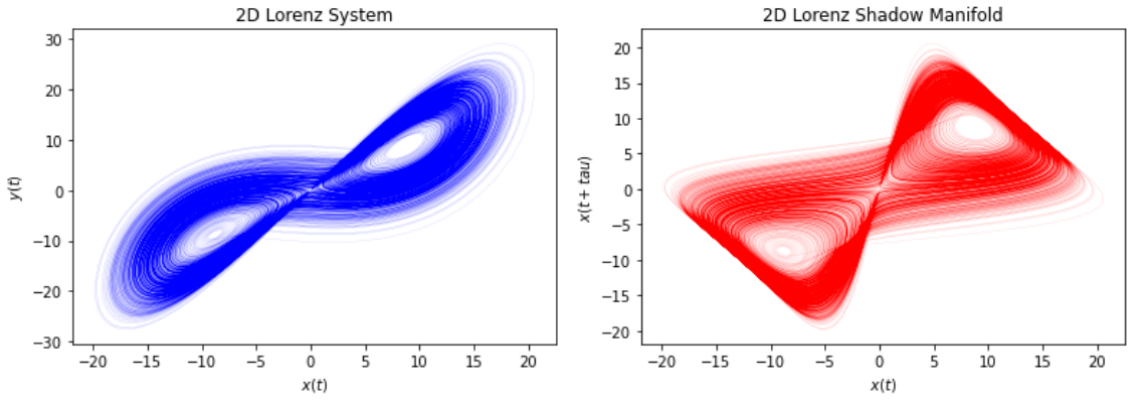


Figure 2.6: Left: 2D Lorenz Manifold, Right: 2D Time Delay Lorenz Manifold ($\mathcal{M}^X$)

Inspired by Takens' theorem is the technique of Convergent Cross Mapping (CCM), which is a method of testing for causality between observables. CCM has the ability to show whether correlations are due to direct interaction between two objects or are due to a shared environment or state influencing both groups simultaneously. With CCM and Takens' theorem as evidence, it is possible to recreate a signal with only partial knowledge of the original system [8]. The paper "Detecting Causality in Complex Ecosystems" goes into more detail about CCM, and can provide a good foundation for this concept [8].

In Figure 2.7, a shadow manifold is shown for both Anode Pearson and Cathode Pearson signals from the HET data. Both were constructed using a delay of $\tau = 150$, which visually produced the cleanest results.

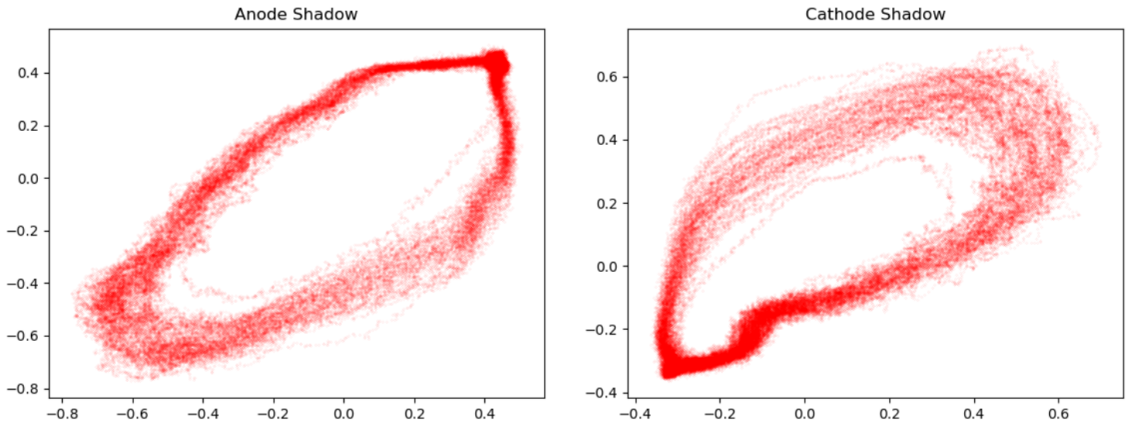


Figure 2.7: Left: Anode Pearson Shadow Manifold, Right: Cathode Pearson Shadow Manifold

Figure 2.8 includes images of the shadow manifold for Anode + Cathode signal and of the shadow manifold for Total Cage singal, again from the provided HET data. These were constructed using a time delay of $\tau = 150$ and were then used throughout the creation of signal reconstructions.

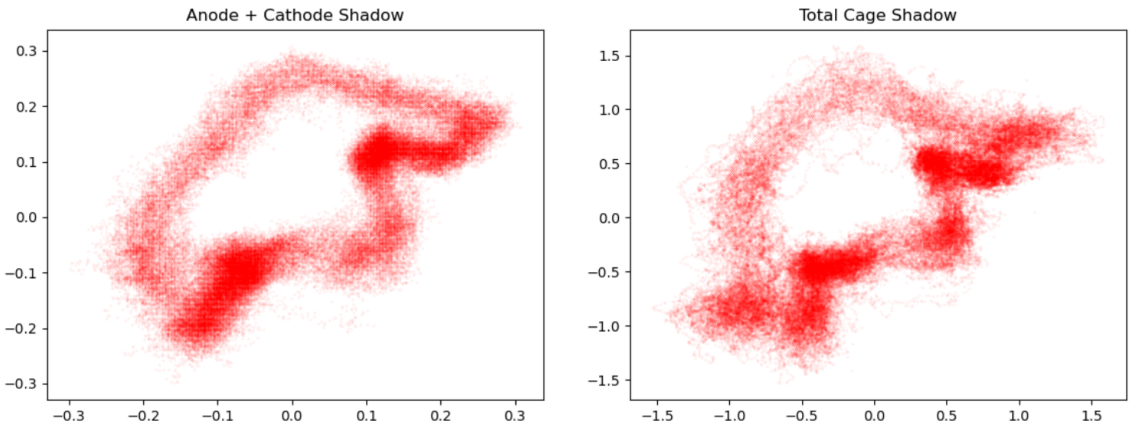


Figure 2.8: Left: Anode + Cathode Shadow, Right: Total Cage Shadow

By using CCM and Takens' theorem and applying them to the given data, it is possible to develop reconstruction methods to denoise a corrupted signal. With this capability, it is possible to learn more about the dynamics of a system like the HET, which contains a certain amount of experimental noise.

2.4 Previous Work with Uniform Mesh

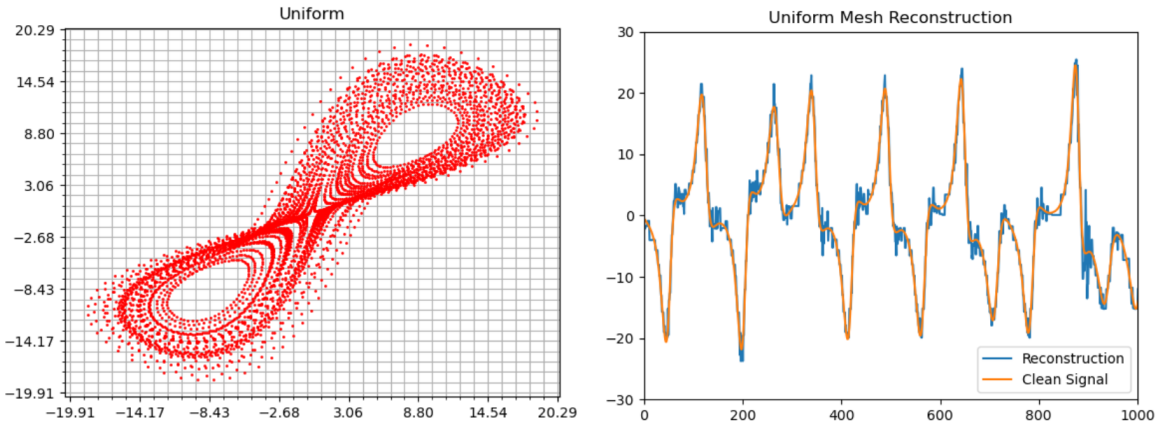


Figure 2.9: Left: Uniform mesh placed over X shadow manifold, Right: Reconstruction of Y signal using uniform mesh.

In AFRL's previous work, the concept of CCM was applied to a uniform mesh-based approach in an attempt to recover signals that have been corrupted by noise. Given a clean observed signal $X(t)$ and a noisy observed signal $\tilde{Y}(t)$ recorded from the same system, this method attempts to recover the uncorrupted signal $Y(t)$ and has been shown to achieve some success in reconstructing some signals, but does not converge with added data as expected and desired. This reconstruction process is described in detail in Reference [6], but is nearly identical to our proposed method in Chapter 3 of this report, aside from the mesh used, the language of programming, and the refinement strategies implemented. Figure 2.9 shows a uniform mesh placed on the time delayed 2D Lorenz manifold, with a signal reconstruction using it.

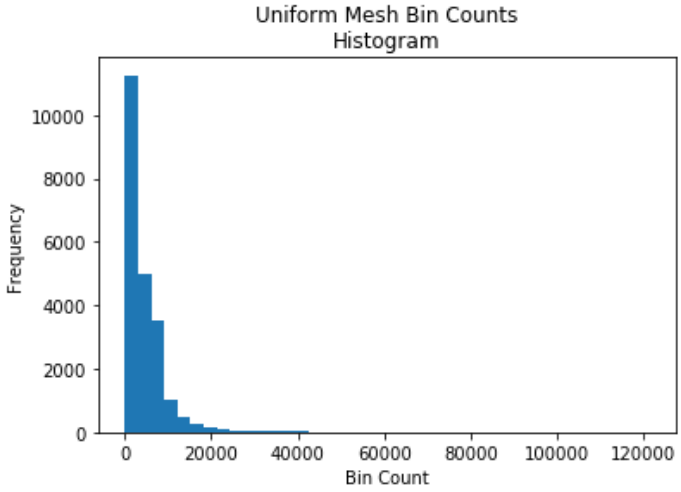


Figure 2.10: A histogram showing the bin counts for the 99.9 million points from the Lorenz X signal on the uniform mesh

It has been shown in Reference [6] that a uniform mesh-based approach can also help recover the signal $Y(t)$ only by knowing the signals $X(t)$ and $Z(t)$, though with certain difficulties such as a wide variation of number of data points per mesh cell and difficulty in

application of refinement strategies. Depicted in Figure 2.10 is a histogram of the number of time-delay embedded data points per uniform cell, which illustrates a considerably uneven distribution, with most cells close to empty and a few cells with over 40,000 data points. This serves as evidence that an update to AFRL's uniform mesh method is greatly warranted.

2.5 Error Quantification

Our specific method of measuring error involves the Pearson Correlation Coefficient (PCC).

It is calculated as

$$PCC = \frac{\sum_{x \in X, y \in Y}(x - \overline{X})(y - \overline{Y})}{\sqrt{\sum_{x \in X}(x - \overline{X})^2 \sum_{y \in Y}(y - \overline{Y})^2}}$$

where X and Y are two real-valued signals.

PCC is used here to provide information to compare the original signal with the reconstruction. Specifically, it is a way to measure the linear correlation between two variables, with a value of 1 representing total positive linear correlation, a value of 0 representing no linear correlation, and a value of -1 representing total negative linear correlation. Our analysis used the value of 1-PCC as a quantification of error, which sufficed as an indication of the signal-to-noise ratio. Note that the signal-to-noise ratio is a measure that compares a signal to the level of noise present. The specific ratio can give information about how much of the original signal is present compared to the noise.

We would also like to note that our results also held when computing the error using the Root Mean Squared Error (RMSE) instead of 1-PCC. Both gave similar information about the reconstructions, and were able to quantify error, though PCC was used more frequently than RMSE.

In general, a uniform mesh and smoothing filters were previously used by AFRL as denoising strategies, but still resulted in imperfect results. It has been hypothesized that these imperfections were largely caused by the use of a uniform mesh. In this project, we implemented an unstructured mesh system which adapted to non-uniform point density so to mitigate these imperfections, and we tested the results of this method.

Chapter 3

Reconstruction Method

In this chapter, we describe our new method of denoising a signal via reconstruction using an unstructured mesh. The reconstruction process was modeled after AFRL's work on reconstruction using a uniform mesh [6]. Before we illustrate this reconstruction method, we will explain our technique for creating the new unstructured mesh.

3.1 Voronoi Diagram

Our nonlinear noise reduction algorithm utilizes an unstructured mesh, known as the Voronoi diagram. We will first define the Voronoi diagram and will then proceed to explain why it is crucial to our algorithms success, as well as how we use it to denoise signals.

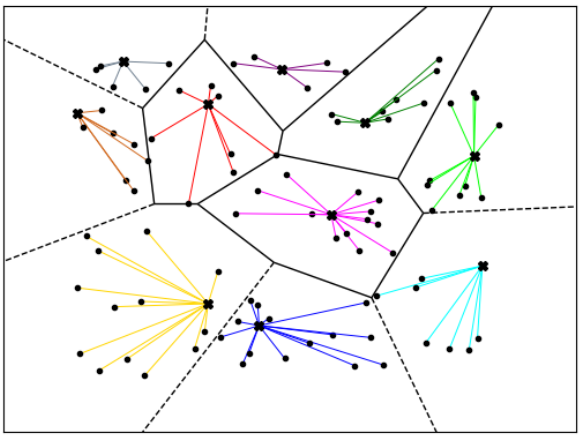


Figure 3.1: Voronoi diagram illustration with Nearest Neighbors strategy.

To construct the Voronoi diagram from a data set, we must first choose a subset of our data to act as *cell representatives*. In Figure 3.1, the cell representatives are the larger data points with an "x" drawn through them. For each remaining data point, we then determine which cell representative the point is closest to with respect to Euclidean distance. We say that particular cell representative is our data point's *nearest neighbor*, and we group all data points with the same nearest neighbor in the same cell. The boundaries of the Voronoi cells

depicted in Figure 3.1 can be thought of as the limit past which that cell's representative is no longer a given data point's nearest neighbor.

In practice, we choose our cell representatives randomly from the time-delay embedded points of a clean signal. In this report, we often refer to the set of cell representatives as simply the *subset*, as it is a randomly selected subset of data points from the shadow manifold. It follows that the number of cells in the mesh is equivalent to the subset size. This subset-selection method causes dense areas of the attractor to be sampled more frequently, resulting in a high volume of small cells where there are more data points. Consequently, the number of data points per cell in the Voronoi diagram will be more evenly distributed than in the previously implemented uniform mesh.

3.2 Signal Reconstruction Algorithm

Our algorithm for nonlinear noise reduction is very similar to AFRL's previous attempts, with the major difference being our unstructed mesh approach. Note that prior to making a proper reconstruction attempt, the time-delay, embedding dimension, timestep, and number of cells parameters must be optimized. The description of these optimization process is presented later in Section 4.1, but for now, we focus on the reconstruction algorithm itself.

3.2.1 Data preparation

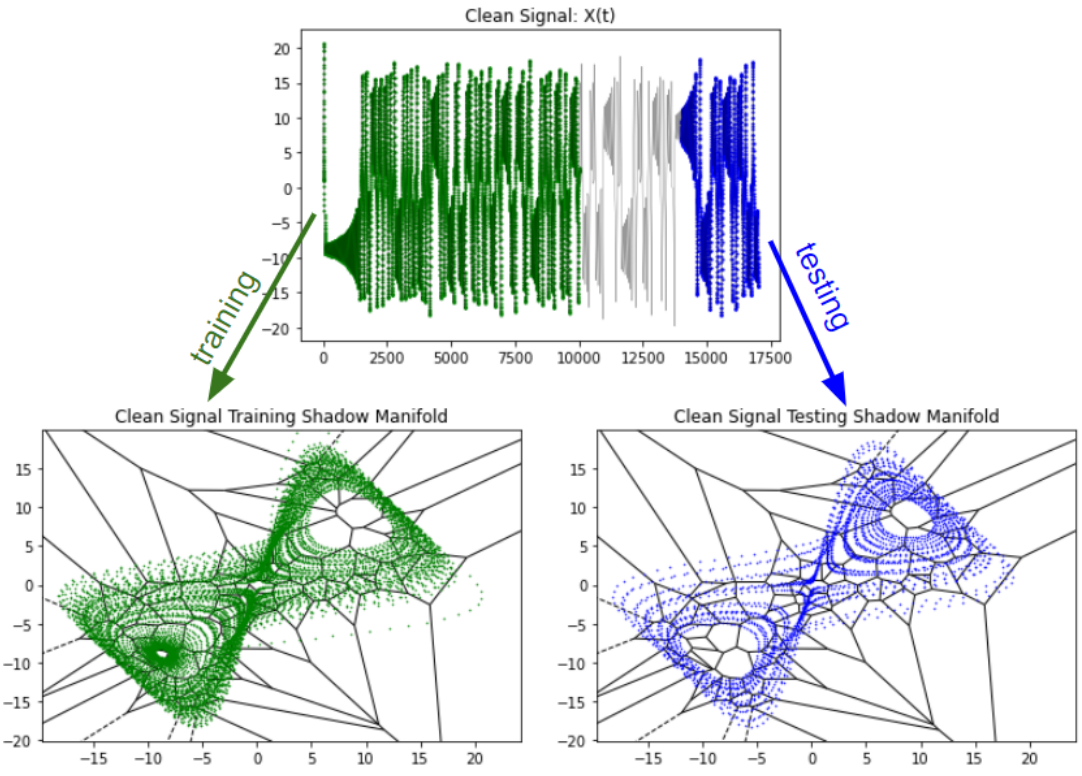


Figure 3.2: Reconstruction Algorithm Part I

The inputs of our algorithm are a clean signal $X(t)$ and a corrupted signal $\tilde{Y}(t)$, both sampled from the same dynamical system over the same time interval and with the same

sampling frequency. We first assign disjoint intervals within the signals $X(t)$ and $\tilde{Y}(t)$ corresponding to training and testing phases, such that the two phases are separated by considerable time. We typically assign these intervals in such a way that plenty of the data is reserved for training. The training data that is used during the algorithm can either be taken to be the entire training interval, or a random sample of points from the training interval, whereas the testing data used must consist of the entire testing interval.

After collecting our training and testing data, we then construct training and testing manifolds from time delays of the signal $X(t)$ in a dimension $d \geq 2$, letting $\mathcal{M}^X_{\text{train}}$ denote the training manifold and $\mathcal{M}^X_{\text{test}}$ denote the testing manifold. After constructing these manifolds, we use a randomly chosen subset of $\mathcal{M}^X_{\text{train}}$ as the cell representatives for a Voronoi diagram, which we place over both $\mathcal{M}^X_{\text{train}}$ and $\mathcal{M}^X_{\text{test}}$. Figure 3.2 illustrates the steps we have taken in the algorithm up until this point, specifically for the Lorenz system with an embedding dimension of $d = 2$.

3.2.2 Training phase

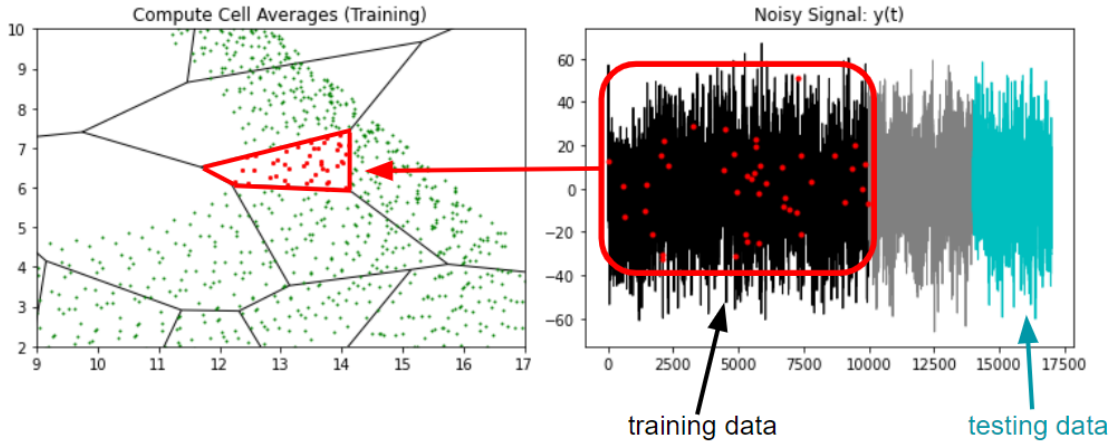


Figure 3.3: Reconstruction Algorithm Part II

In the training phase of the algorithm, we assign an average value to every cell of the Voronoi diagram, using $\mathcal{M}^X_{\text{train}}$ and the training data of $\tilde{Y}(t)$. It is important to note that every data point of the signal $\tilde{Y}(t)$ can be associated with a data point of the shadow manifold generated by $X(t)$ with one-to-one correspondence. Suppose that y_i is a data point satisfying $y_i = \tilde{Y}(t_i)$ for the time t_i. We can then let

$$y_i \mapsto (X(t_i), X(t_i + \tau), X(t_i + 2\tau), \ldots, X(t_i + (d-1)\tau)),$$

where τ denotes the time-delay and d is the embedding dimension. Clearly, the point y_i is mapped into a point on the shadow manifold generated from $X(t)$, allowing us to group together the training points of $\tilde{Y}(t)$ that map into the same Voronoi cell, V_i, of $\mathcal{M}^X_{\text{train}}$. We then compute the average of these points and associate that value to V_i, which will be used to construct a clean signal in the testing phase. The process of grouping together the points of $\tilde{Y}(t)$ that map into the same Voronoi cell of $\mathcal{M}^X_{\text{train}}$ is illustrated in Figure 3.3.

We repeat this procedure for all Voronoi cells, such that we have an average value associated to every cell of the Voronoi diagram. This concludes the training phase of our signal reconstruction algorithm.

3.2.3 Testing phase

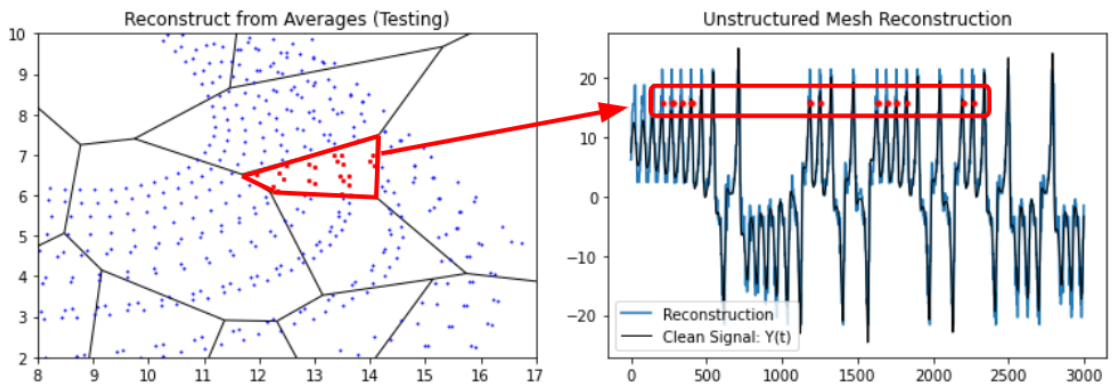


Figure 3.4: Reconstruction Algorithm Part III

We reconstruct the testing data of $\tilde{Y}(t)$ in the following way. Let $\{t_j\}_{j=1}^{N}$ be the collection of times for which the testing data was collected. For each t_j, we then identify the point

$$(X(t_j), X(t_j+\tau), X(t_j+2\tau), \ldots, X(t_j+(d-1)\tau)) \in \mathcal{M}_{\text{test}}^{X}$$

and we determine the Voronoi cell that it resides in. Letting A_j denote the average of that particular Voronoi cell, we set $R(t_j) = A_j$, where $R(t)$ denotes the reconstruction of $\tilde{Y}(t)$. A helpful way to think about this last step is that all of the points that map into the same cell of $\mathcal{M}_{\text{test}}^{X}$ must be assigned the same value in our reconstruction, as is evidenced by Figure 3.4. Repeating this procedure for all Voronoi cells will produces a reconstructed signal, where the signal can take on exactly as many values as there are Voronoi cells being used.

3.3 Method Refinement

After developing the reconstruction algorithm, we implemented strategies to refine it and produce results with higher accuracy and precision. These methods had particular applicability in an unstructured mesh setting, which allowed for further usefulness over AFRL's uniform mesh reconstruction algorithm.

3.3.1 Interpolation

As a modification to the final step of our denoising algorithm, we interpolate between Voronoi averages, so that the reconstruction $R(t)$ can take on a continuum of values, as opposed to just the averages, $\{A_j\}$, of the Voronoi cells. Figure 3.5 motivates the importance of interpolating, as the signals we are attempting to reconstruct are smooth, and the discrete nature of the Voronoi average reconstructions without interpolation are therefore not desirable.

To implement the interpolation between averages, we first construct a *triangulation* of the cell representatives used for the creation of the Voronoi diagram. We experimented with both a Delaunay triangulation and a randomized triangulation, but found that both methods produced similar results. After triangulating the cell representatives, we place the triangulation on top of the data points of $\mathcal{M}_{\text{test}}^{X}$ and assign to each vertex of the triangulation (Voronoi cell representative) the average value of the Voronoi cell that it resides in. This

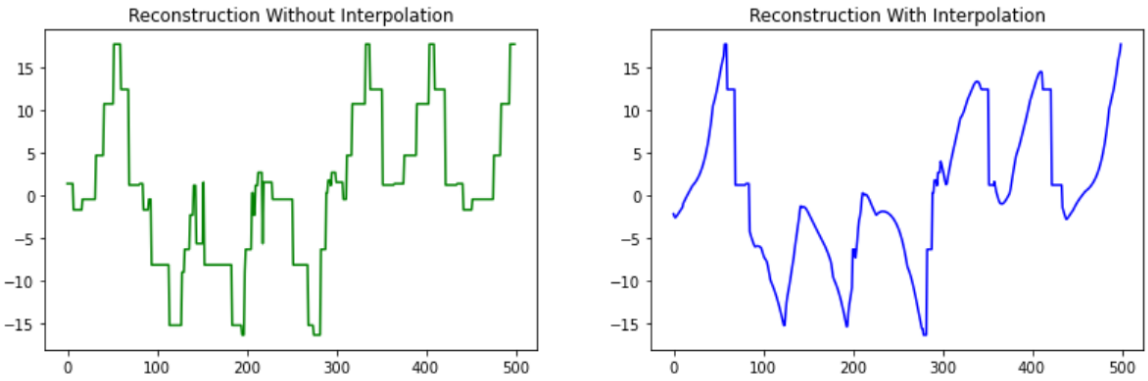


Figure 3.5: Linear Interpolation

algorithm can be generalized to higher dimensions, so we will use the term *simplices* to refer to the generalization of triangles. For every point $p \in \mathcal{M}^X_{\text{test}}$ that lies within the convex hull of the cell representatives, it can be shown that p also resides in one of the simplices. The *barycenter* of that simplex can be thought of the center of mass, given by the weights of the Voronoi averages associated with each vertex, and a *barycentric interpolation* can be viewed as a generalization of linear interpolation between these values. We conduct a barycentric interpolation between the vertices of that simplex to determine exactly what value the point p corresponds to in the signal reconstruction. For points that lie outside of the convex hull of cell representatives, we simply use the average value associated to the Voronoi cell that they reside in for the signal reconstruction at that point.

3.3.2 Cell Adaptation: k-Means Clustering

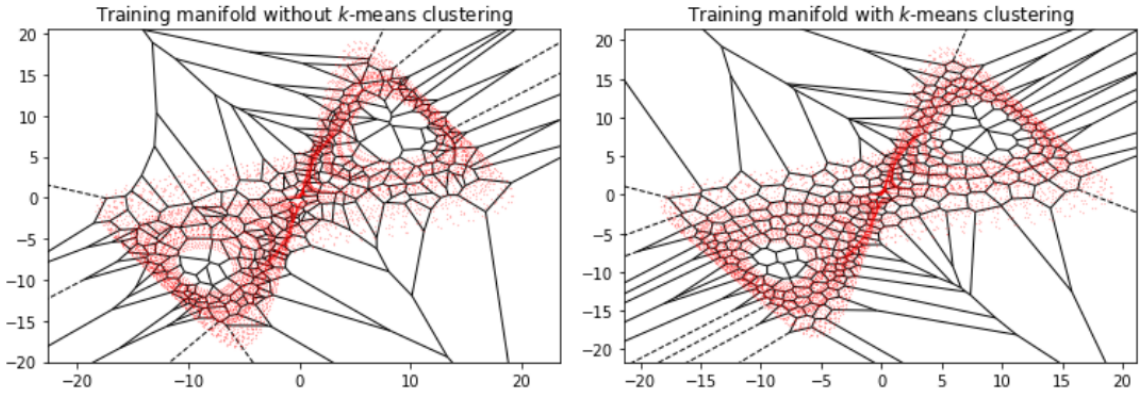


Figure 3.6: Effect of k-means clustering on Voronoi diagram

To improve the choice of the random sample for use in constructing the unstructured mesh, we also tried running the k-means clustering algorithm on the training data set. The k-means clustering algorithm iteratively chooses new cell representatives by identifying the center of mass for each Voronoi cell until the algorithm converges to a solution. We elected to use minibatch k-means to improve processing time. Figure 3.6 shows the visual impact of applying the k-means clustering algorithm, while Figure 3.7 highlights the difference in point distribution while using k-means. In Figure 3.7, we fit Gamma distributions to the

histogram plots and note that the distribution is more skewed without the application of k-means clustering.

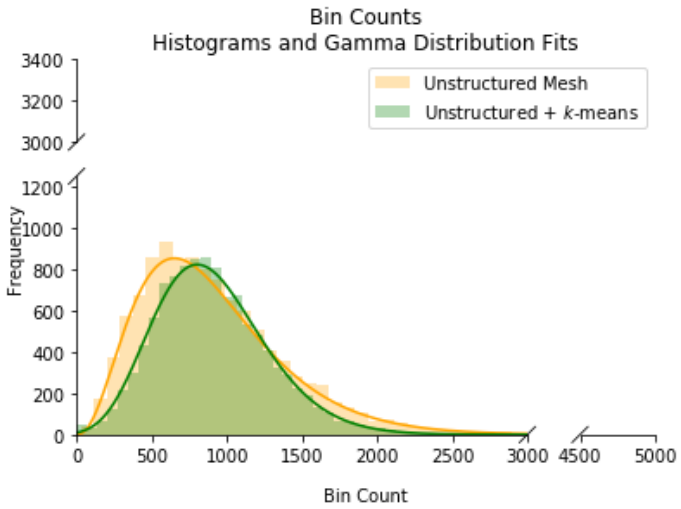


Figure 3.7: Effect of k-means clustering on histogram bin counts

In Figure 3.8, we show the effect of interpolation and k-means clustering on the locations of error. These plots detail where on the testing shadow manifold the areas of higher error correspond to in the reconstruction, with yellow indicating the highest errors and purple indicating the lowest errors. In Figure 3.8 we use reconstructions of the Lorenz Z signal from the X signal with 1,000,000 training data points, $\tau = 17$, and a sample size of 300. While the effects depicted in Figure 3.8 could change for different simulation parameters, we note here that by itself k-means clustering is not significantly impactful, but with the addition of interpolation, it makes a significant contribution to error reduction. Aside from the long run-time associated with the k-means algorithm, we see that both k-means clustering and interpolation are helpful tools for reducing the error in our reconstructed signals.

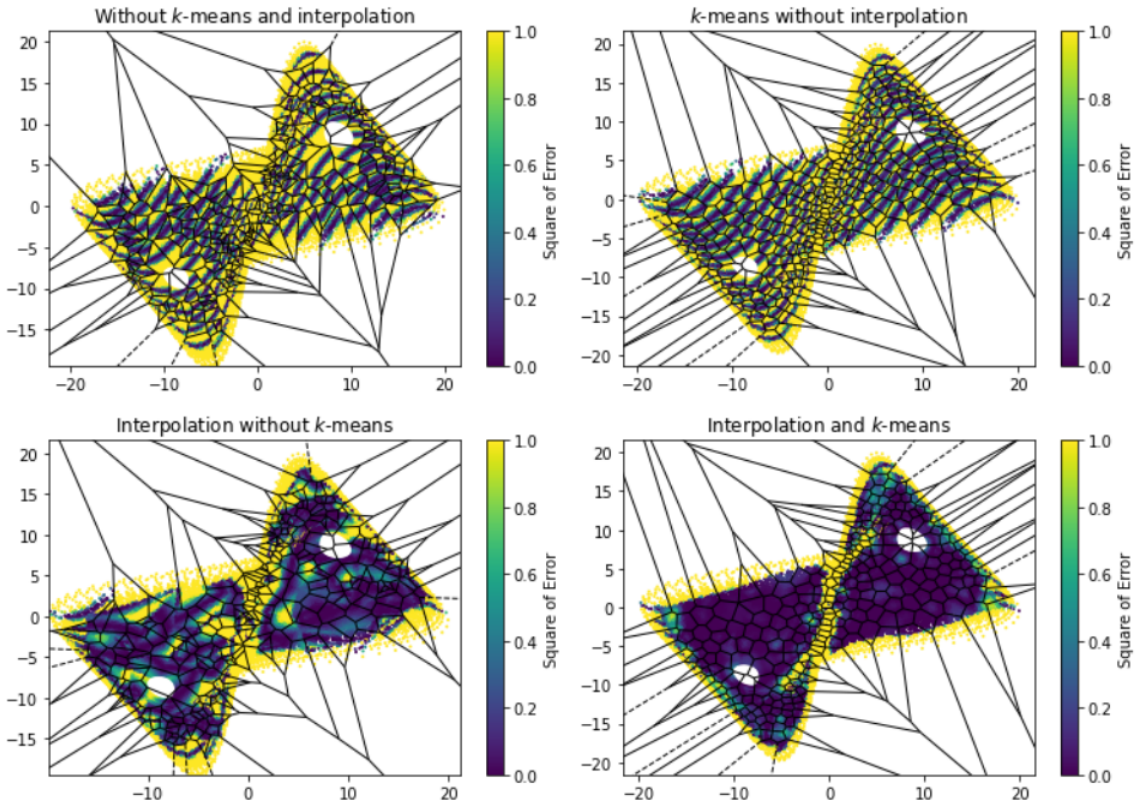


Figure 3.8: Effect of linear interpolation and k-means clustering on reconstruction error

Chapter 4

Evaluation of Method

With the reconstruction method in hand, our goal was to reconstruct a signal with low error, which we quantified using the Pearson Correlation Coefficient (PCC). Alongside testing our method, however, we also worked to optimize the parameter values used in the algorithm. These parameter values include the timestep used in the ODE solver, the time delay τ, the embedding dimension d, the subset size, and the amount of training data. This chapter illustrates the methods we used to judge the success of our reconstruction algorithm and optimize parameter values.

4.1 Parameter Optimization

Before we attempted a reconstruction, we analyzed the effects that different parameters would have on each set of data. This section discusses our methods of parameter optimization, roughly following the order of which the parameters were used in the algorithm.

4.1.1 Timestep parameter

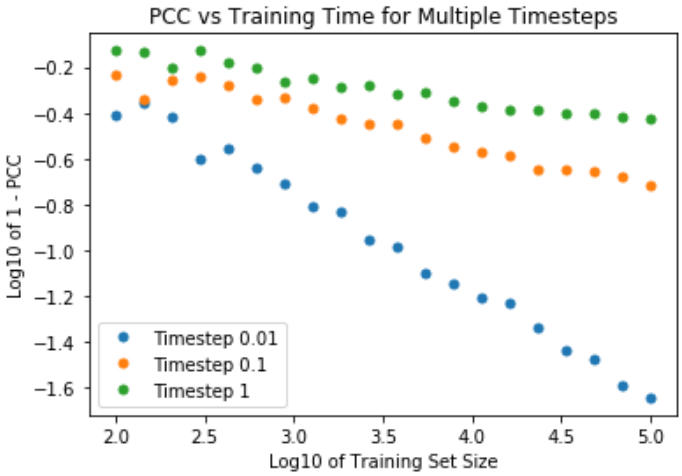


Figure 4.1: Increasing the timestep causes the convergence to break.

The data in Figure 4.1 reconstructed the Lorenz Z signal from the Lorenz X signal. We observed that if the timestep is too high for the ODE solver, then the algorithm will

not converge; this is due to the ODE solver breaking. We can demonstrate that this is why the algorithm does not converge because if we generate the data with a small timestep, but then extract from this only every 10th or 100th element, we still get converging log-log plot curves. We also expect non-convergent behavior for experimental data sampled with too large a timestep, and thus graphing curves like this can help inform experimenters as to what sampling rate they should choose.

4.1.2 Time Delay τ

We begin our algorithm with a parameter search for the optimal τ and d values.

To find a good estimate τ^* of the time delay τ, we find the first local minimum of the Average Mutual Information (AMI) of the signal X as a function of τ following the method proposed by [16]. In particular, we calculate the AMI for a time series $X(t)$ where $0 \leq t < n$ by first binning the values of $X(t)$ into m uniform bins, where $m = 1 + \log_2(n - \tau) + 0.5$ in accordance with an algorithm developed by Alexandros Leontitsis [18]. Next, we compute

$$\sum_{ij} p_{ij} \log_2 \frac{p_{ij}}{q_i r_j}$$

where p_{ij} is the fraction of values $0 \leq t < n - \tau$ such that $X(t)$ is in bin i and $X(t + \tau)$ is in bin j, q_i is the fraction of values $0 \leq t < n - \tau$ such that $X(t)$ in in bin i, and r_j is the fraction of values $0 \leq t < n - \tau$ such that $X(t + \tau)$ in in bin j. We then can plot the AMI as a function of τ and use the first local minimum of the resulting function as a heuristic for τ^*. For the rest of the report, we refer to our chosen $\tau*$ simply as τ.

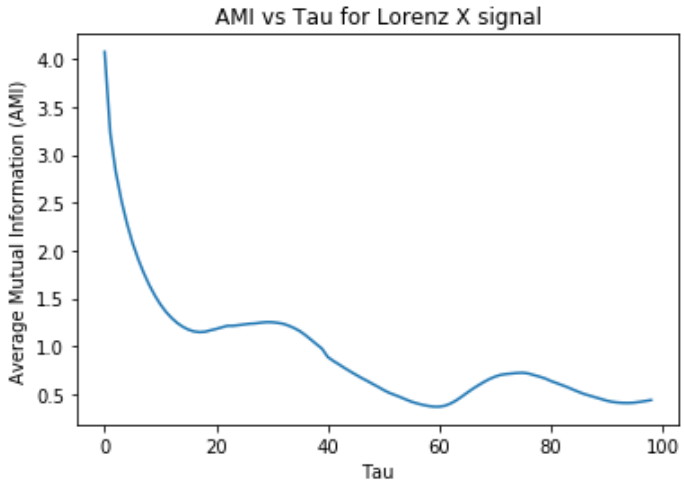


Figure 4.2: Average Mutual Information (AMI) Graph for the Lorenz X signal

In Figure 4.2, we can see that the first minimum falls at approximately $\tau = 17$, so this is a good guess for the optimal τ for the Lorenz X signal. Similarly, for the HET Anode+Cathode signal, we see in Figure 4.3 that there is a "corner", almost a local minimum, around $\tau = 150$, while the first local minimum occurs closer to 200. The optimal value should be somewhere around this area.

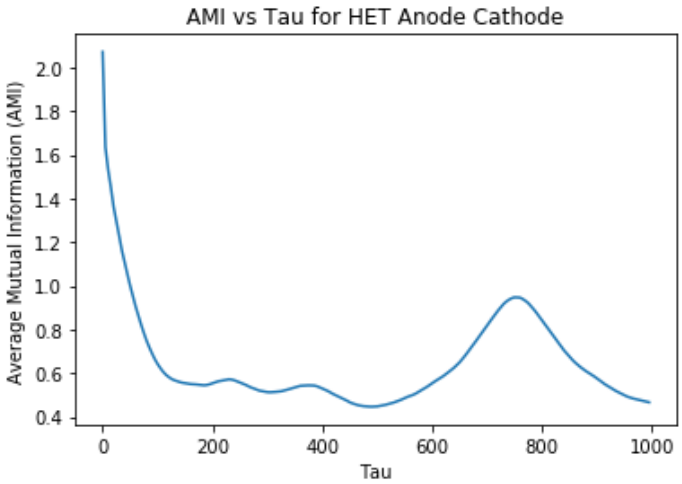


Figure 4.3: Average Mutual Information (AMI) Graph for the HET Anode+Cathode signal

In addition to using the AMI method, we also qualitatively found which τ values caused the reconstructed shadow manifold to minimize the number of crossing regions. Different values of τ can cause the shadow manifold to look completely different, and some values can make reconstructions much worse. In Figure 4.4, the impact of τ on the Lorenz system is illustrated. Here, we find that $\tau = 10$ looks best out of these options.

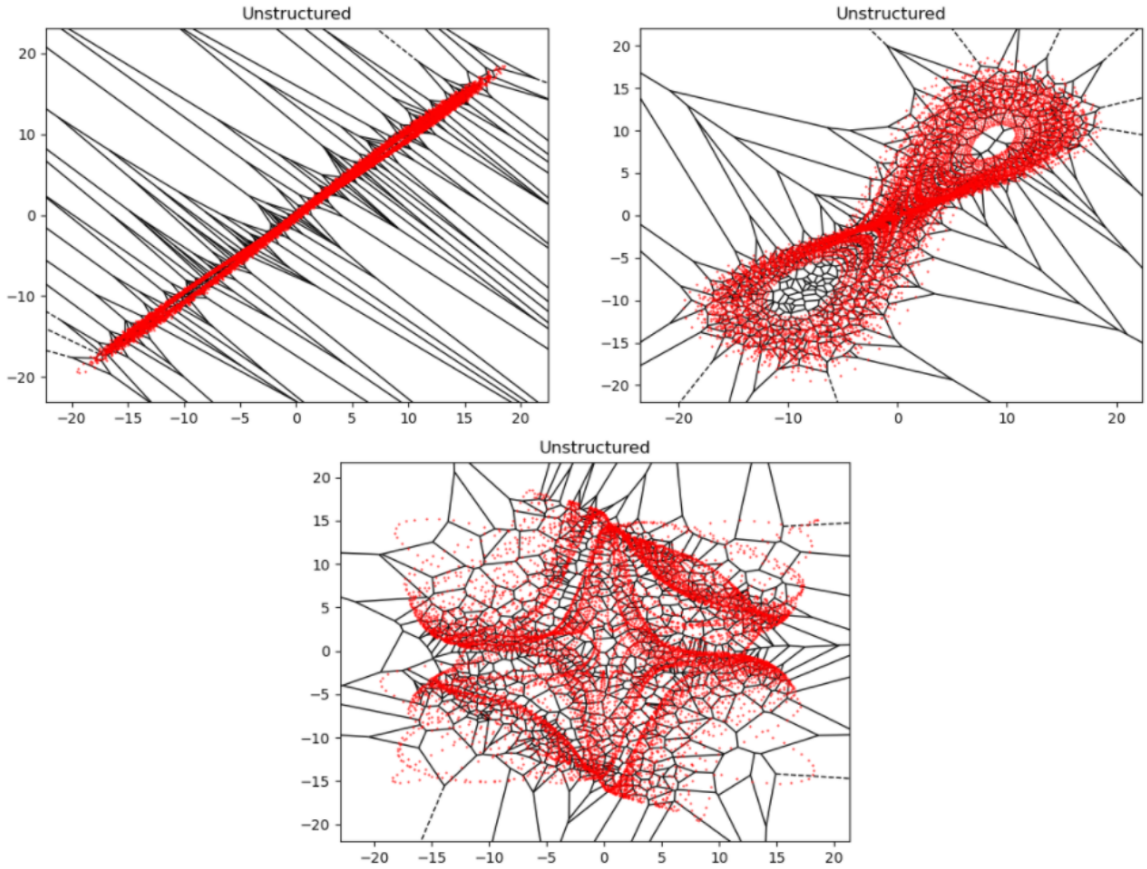


Figure 4.4: Varying value of τ: 1, 10, and 100, respectively

The choice of τ remains important for the HET data, especially because the data tends to contain several crossing points which cause issues in the reconstructions. In Figure 4.5, three different values for τ are shown along with the corresponding shadow manifold. Specifically, these are all portraits of the Anode + Cathode HET data shadow manifold, with standardized parameters other than τ. For instance, these are all shown with a subset size of 300. In most of the reconstructions, the value of τ was chosen to be 150 from both visual analysis of the fit and from tests showing the error tended to be lower with that value as illustrated further in Section 4.4.

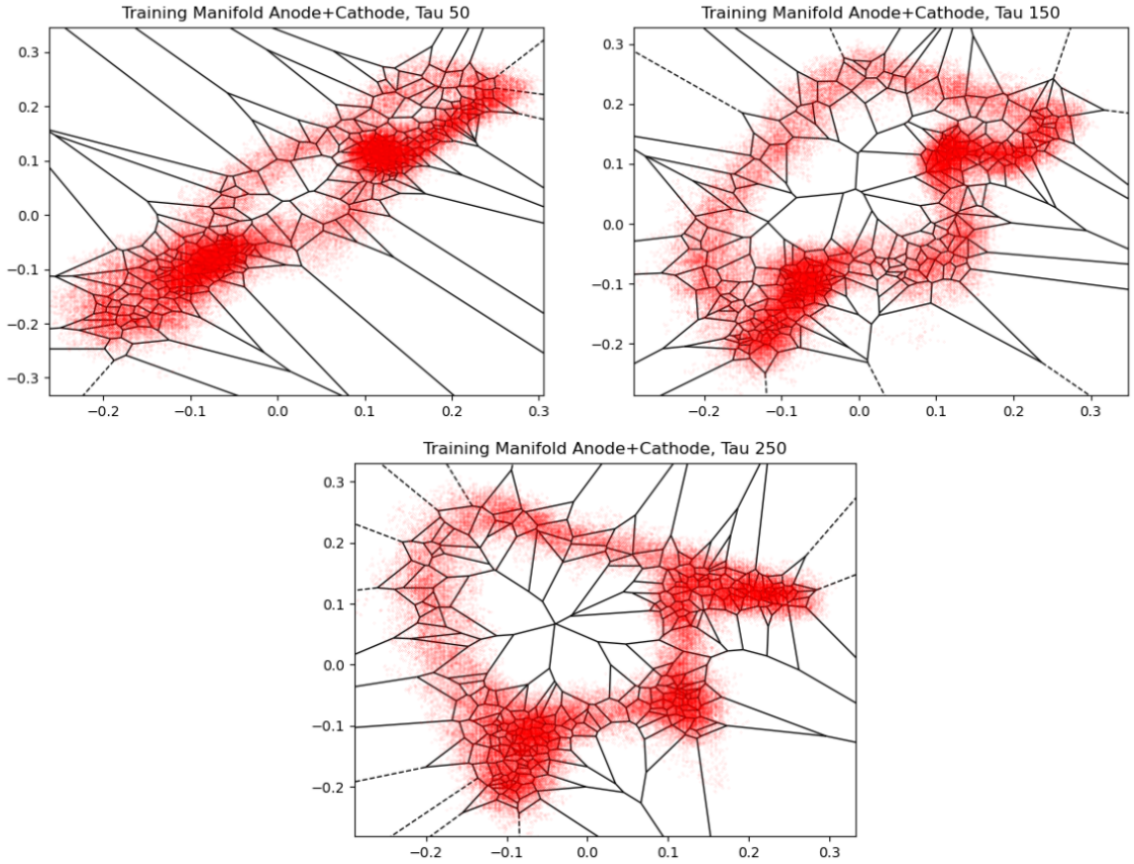


Figure 4.5: Varying values of τ, Anode + Cathode Shadow Manifold

In Figure 4.6, the shadow manifold for the Total Cage data is shown with a few different values of τ. This is another piece of real data used in later reconstructions. Out of the three options for τ presented in these images, we find that $\tau = 150$ looks relatively good. Indeed, by viewing the shadow manifold over a wide range of values of τ, we find that the first region where the shadow manifold appears to have minimal crossing areas is between 150 and 200.

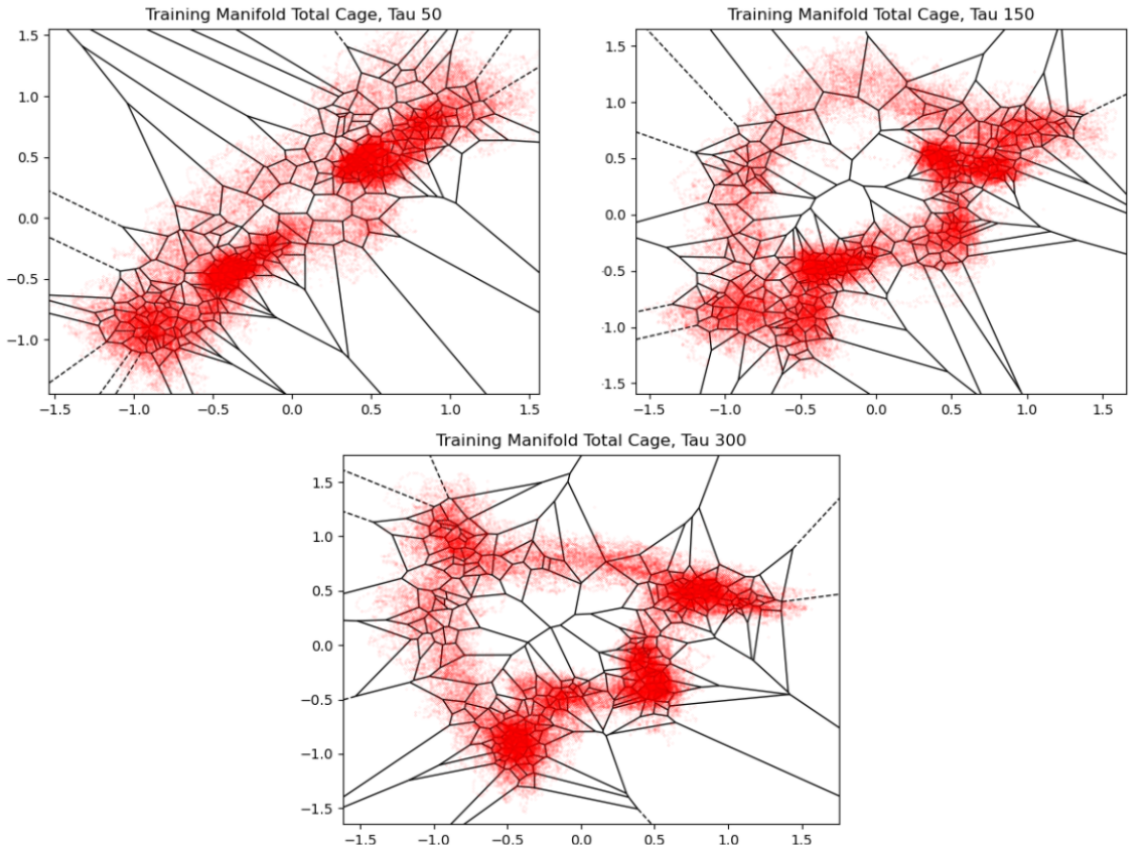


Figure 4.6: Varying values of τ, Total Cage Shadow Manifold

4.1.3 Embedding Dimension d

After having selected a sufficient value for τ, we try to optimize the embedding dimension for the shadow manifold.

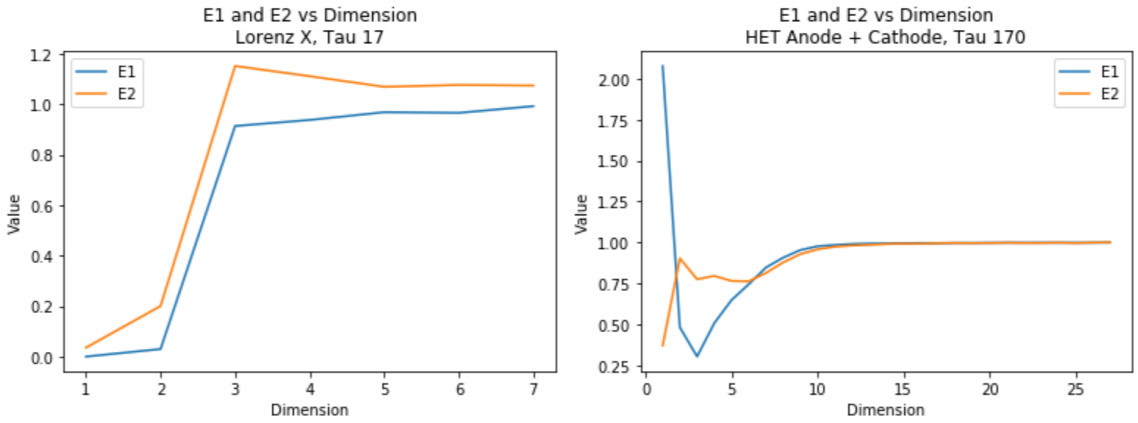


Figure 4.7: Plots of E_1 and E_2 according to Cao's method for the Lorenz X signal on the left and for HET Anode+Cathode on the right.

In practice, the minimal embedding dimension depends heavily on the selected value for τ, and we select it using Cao's method [17]. Cao's method describes a way to produce values $E1(d)$ and $E2(d)$, and it says that the minimal embedding dimension is approximately the

dimension d at which $E1(d)$ and $E2(d)$ stop changing. Whether it is better to follow the the heuristic provided by $E1$ or $E2$ depends on how noisy the data is, but because we only applied Cao's method as a rough heuristic in this work, we will leave the details to Cao's paper.

In Figure 4.7, we see plots that show a heuristic for the minimal embedding dimension d_{min}. We see that the minimal embedding dimension is 3 for the Lorenz X signal, while it is at least 10 for the HET Anode+Cathode signal.

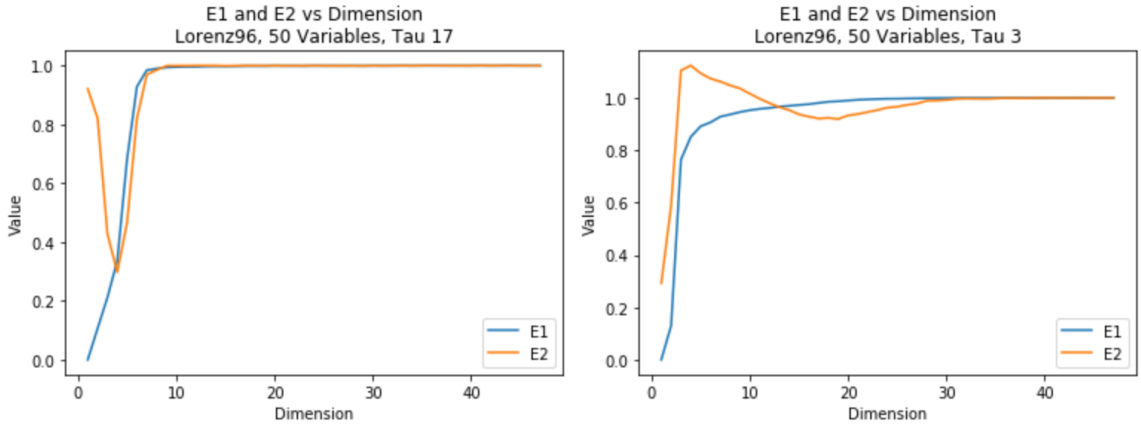


Figure 4.8: Plots of E_1 and E_2 according to Cao's method for the Lorenz 96 system.

In Figure 4.8, we see how fragile this process really is. By varying τ from 17 to 3, the predicted minimal embedding dimension varies from around 10 to around 30. Therefore, Cao's method only provides a heuristic for our algorithm, and the optimal choice of d should be sought based on observed results of the algorithm.

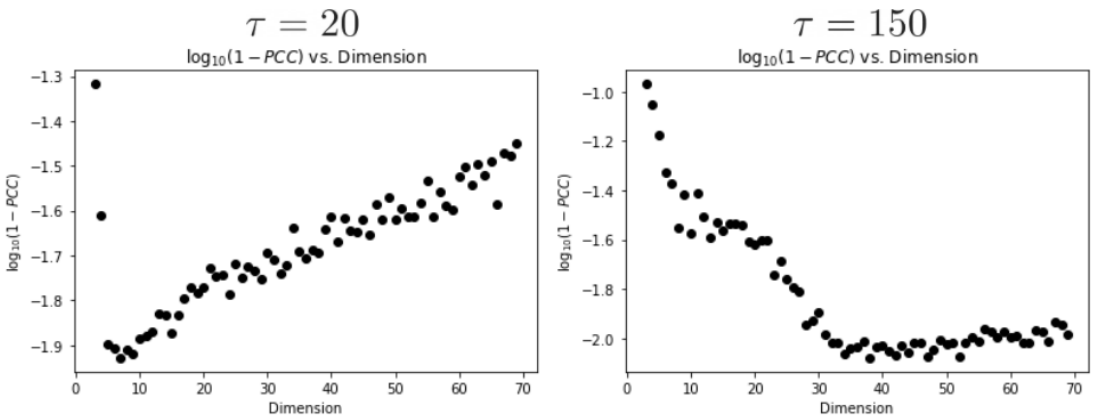


Figure 4.9: Anode+Cathode Reconstruction Error by Dimension

We examined Cao's predicted choice of d for the HET Anode+Cathode signal by plotting a reconstruction error of the Cathode Pearson signal vs. dimension plot, for a fixed sample size and amount of training data. As expected, we see in Figure 4.9 that the optimal value of d depends heavily on the chosen value for τ, as a τ of 20 corresponds roughly to an optimal embedding dimension of 35, whereas a τ of 150 corresponds to an optimal embedding dimension of roughly 7. It is interesting to note that the AMI indicates $\tau = 150$ to be optimal, and though this value of τ clearly corresponds to a lower optimal embedding

dimension than $\tau = 20$, its lowest absolute error is greater than the case where $\tau = 20$. This tells us that solely using the AMI and Cao's method to determine τ and d is not sufficient, and that attention must also be payed to experimental results attempting to optimize the algorithm's performance.

Although it appears like our AMI method is somewhat unstable, there has been research into better methods for finding the optimal τ [14].

4.1.4 Number of Cells

The optimal number of cells in the Voronoi diagram - which we often refer to as the "subset size" - depends on many of the simulation parameters, chiefly the amount of training data available and the amount of noise that our signal has been corrupted with. If too few cells are used then the reconstructions are unable to approximate the true dynamics of the system from which our signals have been sampled from, and if too many cells are used then our algorithm is unable to filter out large amounts of noise. This indicates that to observe success in our reconstruction, one must let the number of cells increase as the amount of available training data increases.

To evaluate the optimal number of cells, we examined several error plots of reconstructions using a fixed numbers of cells across varying amounts of training data. Because of its direct connection to the evaluation of reconstruction convergence, we leave the details and figures of this process to Section 4.3.

4.2 Successful Reconstructions

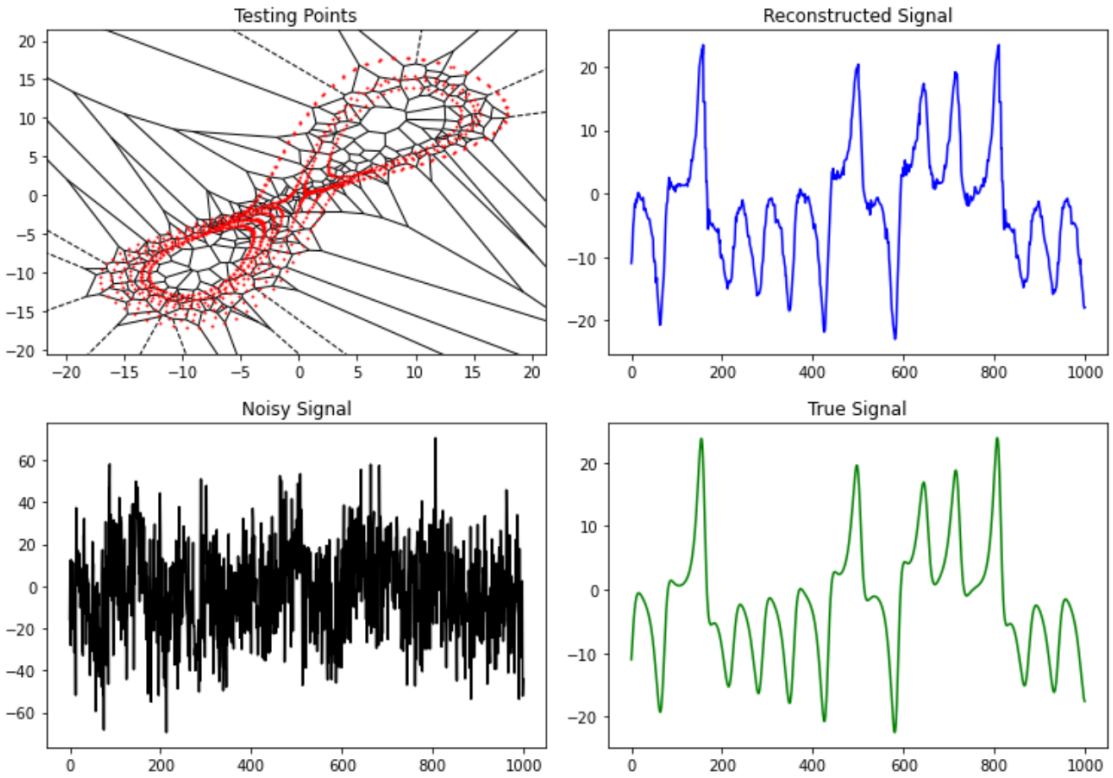


Figure 4.10: Lorenz System Reconstruction

Perhaps the most exciting event of this project was to see successful reconstructions of time series produced by our method. In this section, we offer several illustrations of these reconstructions, which provide useful initial evidence that our method was a success.

4.2.1 Test Data

We initially applied our denoising algorithm to the Lorenz, Rossler, and Chen systems to visually examine its success in reconstructing signals before quantifying its error and searching for convergence. Throughout, we use the time series $X(t)$ to reconstruct the time series $Y(t)$ with added Gaussian noise. Beginning with the Lorenz system, Figure 4.10 shows the points on the testing manifold, the corrupted signal, the reconstruction, and the true signal.

In addition to the Lorenz system, the Chen and Rossler systems were also used to test the effectivness of the denoising algorithm. The Chen system differential equations are listed below, where we used parameters $a = 40, c = 28$, and $b = 3$. The initial conditions are: $X(0) = -0.1, Y(0) = 0.5$, and $Z(0) = -0.6$.

$$\begin{aligned}\frac{dX}{dt} &= a(Y - X)\\ \frac{dY}{dt} &= (c - a)X - XZ + cY\\ \frac{dZ}{dt} &= XY - bZ\end{aligned}$$

After producing a reconstruction from the Lorenz system, we moved on to examine the Chen system, with the parameters as stated. We once again found success in reconstructing a time series with substantial synthetic noise added to it, as depicted in Figure 4.11. This is a reassuring result, as it shows the ability of the method to work on different systems.

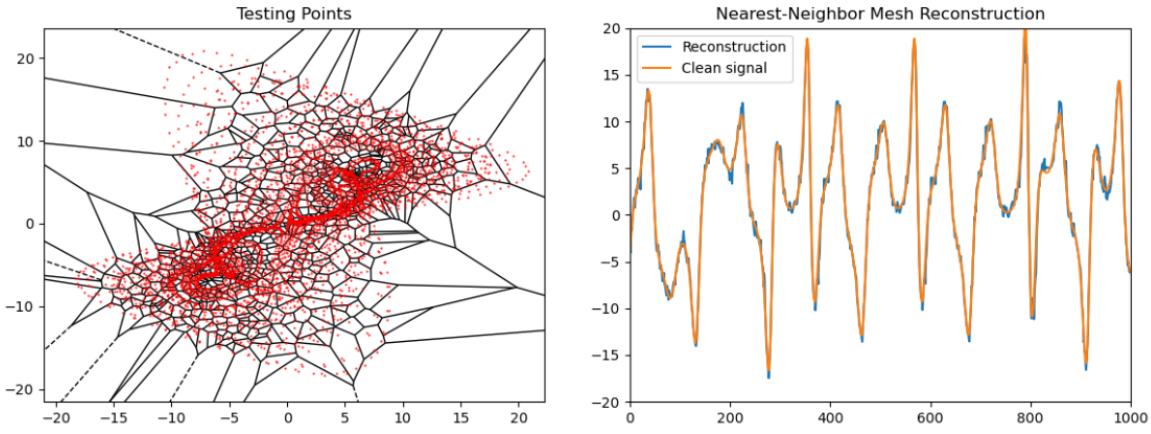


Figure 4.11: Chen System Reconstruction, $\tau = 10$

The Rossler system differential equations are shown below, where we used parameters $a = 0.38, b = 0.35$, and $c = 4.5$. The initial conditions are: $X(0) = 2.0, Y(0) = 2.0$, and $Z(0) = 2.0$.

$$\frac{dX}{dt} = -Y - Z$$
$$\frac{dY}{dt} = X + aY$$
$$\frac{dZ}{dt} = b + Z(X - c)$$

Using the Rossler system, we initially found it more difficult to achieve a satisfactory signal reconstruction. Figure 4.12 shows our initial attempt of a signal reconstruction from the Rossler system.

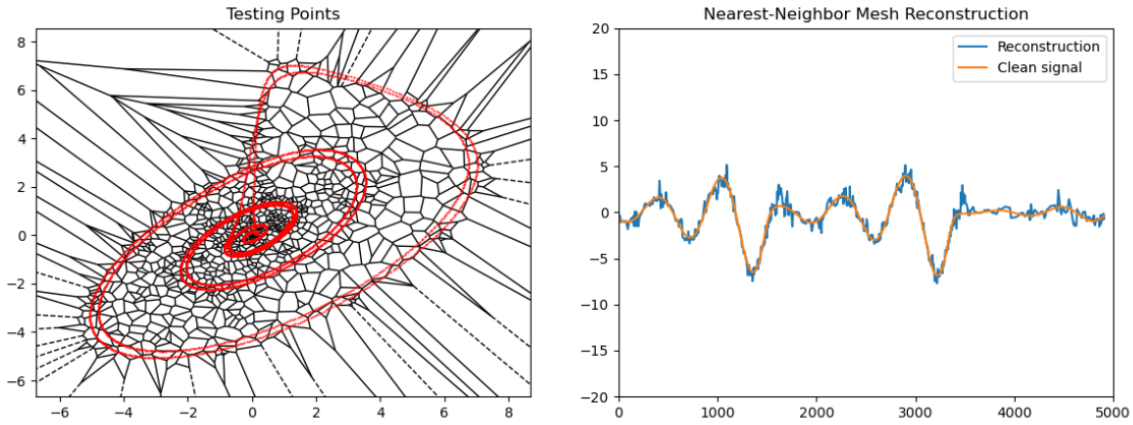


Figure 4.12: Rossler System Reconstruction I, $\tau = 90$

The Rossler system is a three-dimensional manifold, so the apparent errors may be due to the fact that we only performed time-delay embeddings in two dimensions. It is also possible that we needed to increase the amount of training data available. Furthermore, the amplitude of the signals from the Chen and Lorenz systems is on average larger than the amplitude of signals from the Rossler system, and since we have been applying Gaussian noise with a standard deviation of 20, it would make sense to scale this quantity by the relative "size" of our signals. Therefore, a more appropriate standard deviation of noise for the Rossler system would certainly be smaller. In Figure 4.13, we show another attempt at reconstructing a signal from the Rossler system after reducing the standard deviation of the added noise to 10, embedding in three dimensions, and increasing the training time. As seen in the figure, these changes considerably improved the reconstruction.

We then applied the method of reconstruction to test data, in order to examine its success with real world information. The specific data provided was as clean as possible, in order to gather more information about how the method works with real information. So, for some reconstructions, a small amount of noise was added to the signal in order to test effectiveness.

In Figure 4.14, noise was added to the Cathode Pearson signal provided by AFRL at a scale of 0.25. Then, we show a reconstruction compared to the provided signal. The original provided signal is shown in Figure 2.1. The shadow manifold used in this reconstruction is from Anode+Cathode data, seen in Figure 4.5. Specifically, the dimension used to create this reconstruction was 30, along with a time-delay of $\tau = 20$.

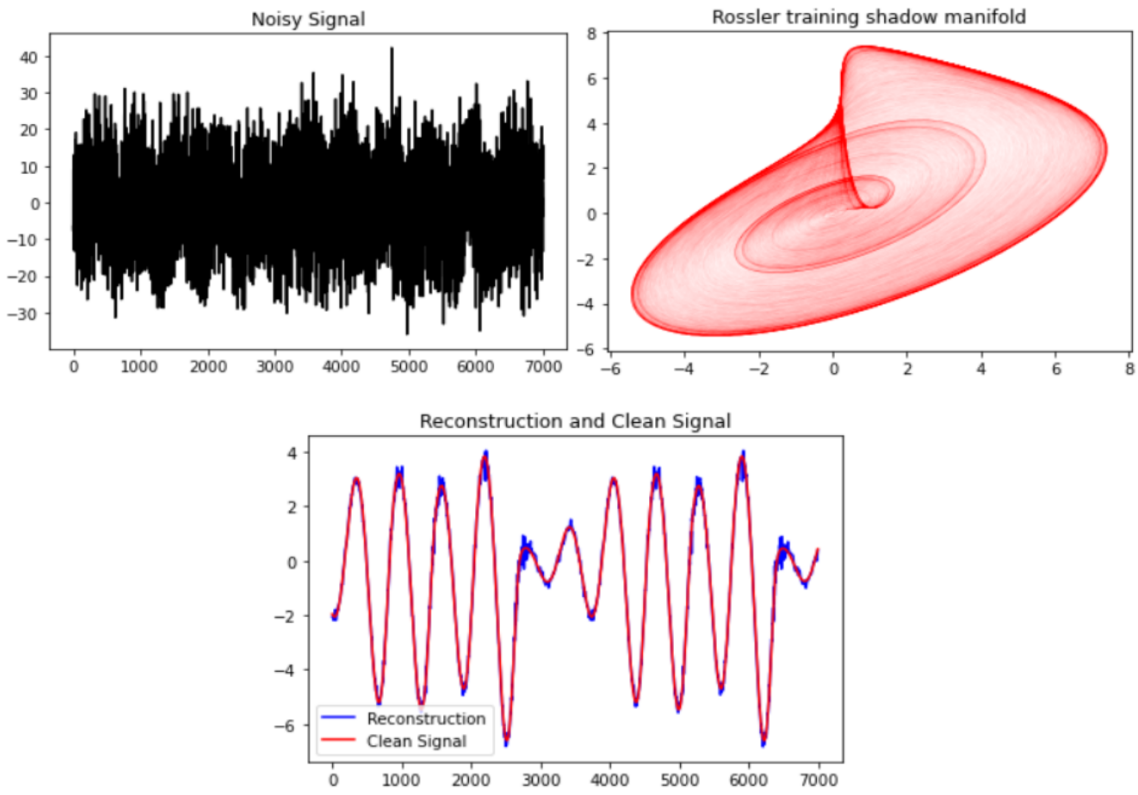


Figure 4.13: Rossler System Reconstruction II, $\tau = 90$

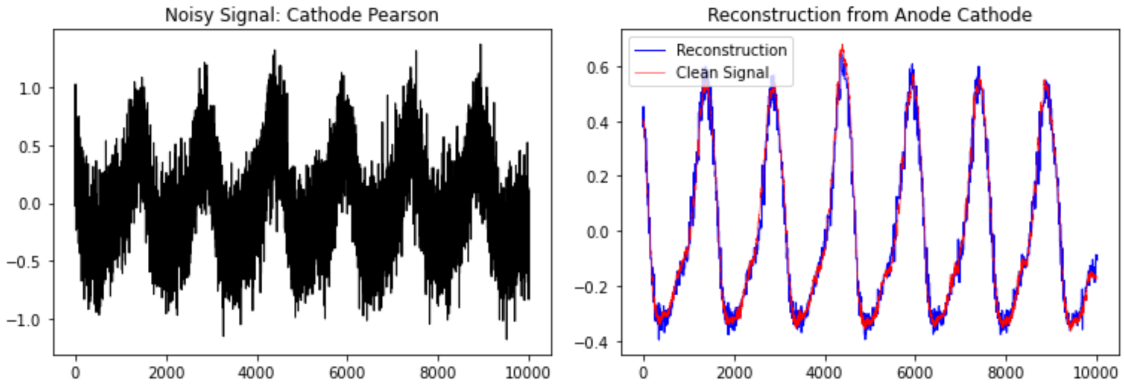


Figure 4.14: Left: Noise Added to Cathode Signal, Right: Cathode Signal Reconstruction from Anode+Cathode

4.3 Error Convergence

After seeing visual evidence that our algorithm is successful via our reconstruction plots, we aimed to quantify success. In this section, we describe what we look for in terms of error convergence in both situations where we reconstruct known signals with synthetic noise added and unknown signals with original noise.

4.3.1 Convergence to a Known Clean Signal

We identify convergence, or success of our reconstruction, as a linear log-log plot of 1-PCC vs. the number of data points used for training. This would indicate exponential decrease in the reconstruction error, and so as the amount of training data approaches infinity, the error should likewise approach 0.

It is important to note that the success of our reconstructions depends heavily upon the number of cells used for the Voronoi diagram, as indicated in Section 4.1.4. Moreover, the optimal number of cells depends on many of the simulation parameters, chiefly the amount of training data available and the amount of noise that our signal has been corrupted with. If too few cells are used then the reconstructions are unable to approximate the true dynamics of the system from which our signals have been sampled from, and if too many cells are used then our algorithm is unable to filter out large amounts of noise. This indicates that to observe optimal convergence one needs to let the number of cells increase as the amount of available training data increases. To determine this optimal relationship, we first plotted log(1-PCC) vs. the amount of available training data for a fixed number of cells, which we varied by factors of two.

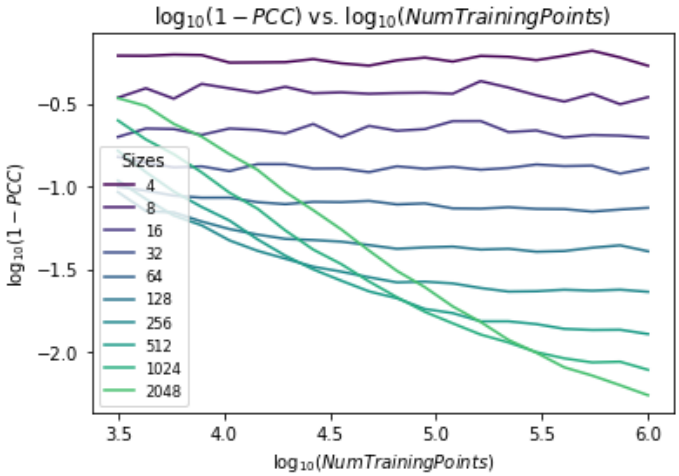


Figure 4.15: Fixed sample size plots for Lorenz reconstructions.

Shown in Figure 4.15 is a fixed sample size convergence plot for the Lorenz system, in which the X signal reconstructs a noisy Z signal with an embedding dimension of 3. Gaussian noise with a standard deviation of 15 was applied to the Z signal, which corresponds to a signal-to-noise power ratio of roughly 1.3. Note that every data point that our fitted curves in Figure 4.15 was taken to be an average of roughly 20 simulation trials. From this fixed sample size plot, there are several important features to note. The convergence line for a given sample size eventually levels out entirely, and in the limit of large amounts of training data the difference between log(1-PCC) for each of the curves appears to be equal. Most importantly, it appears that a tangent line can be drawn on the plot to indicate the optimal reconstruction error for our training interval, given the sample sizes we have tested. Figure 4.16 depicts this tangent line by emphasizing the data point for each training time with minimal reconstruction error. The regression line for the points highlighted in 4.16 would be tangent to the curves for several distinct sample sizes and could therefore help us optimize the relationship between the sample size and amount of training data. Furthermore, the crossing points between these curves near the tangent line seem to occur at

evenly spaced intervals, suggesting that the relationship between the optimal sample size and the amount of training data could be relatively straightforward.

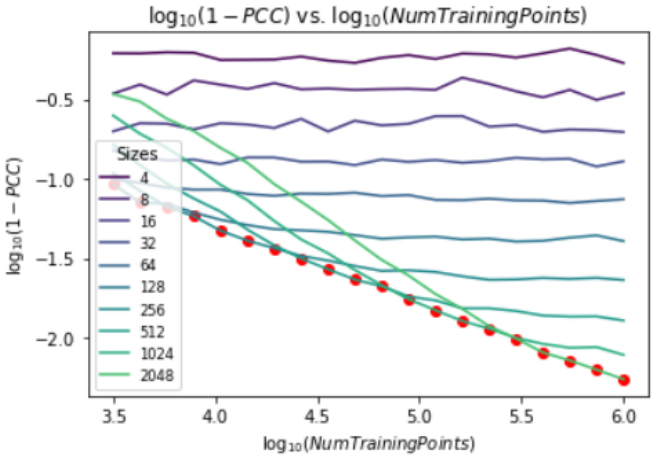


Figure 4.16: Optimal error for fixed sample size plots for Lorenz reconstructions.

Given the evenly spaced curve intersections near the aforementioned tangent line and the fact that our sample sizes were tested in powers of two, it would make sense that the logarithms of amount of training data and optimal sample size would have a linear relationship. Figure 4.17 explores this relationship and confirms that the logarithms of the optimal sample size and amount of training data do indeed vary linearly.

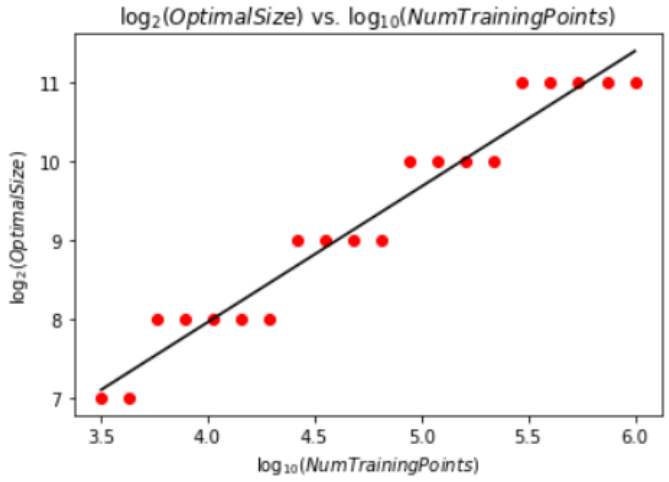


Figure 4.17: Relationship between sample size and amount of training data

For this particular case, we found that the sample size depended on the amount of training data by

$$\log_2(Size) = 1.72 \cdot \log_{10}(Num) + 1.08 \tag{4.1}$$

where $Size$ is the number of cell representatives used to construct the Voronoi diagram, and Num is the amount of available training data.

Using equation 4.1 to parameterize the sample size by training time produced the convergence plot in Figure 4.18. With a slope of -.495, a negligible p-value, and an R^2 value of .993, this optimized log-log plot indicates clear statistical convergence.

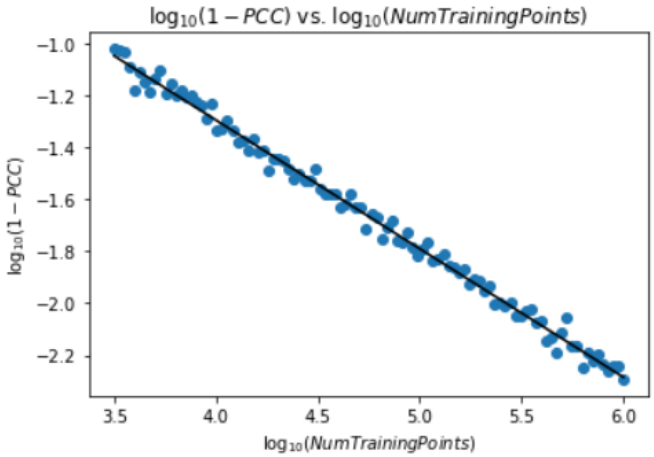


Figure 4.18: Lorenz convergence for varied sample size (I)

Figure 4.18 explores the convergence properties of our algorithm as applied to the Lorenz system for a sample size that was varied by training times over an interval for which we were sure of the relationship between the two parameters. However, if the convergence plot remained perfectly linear using the same relationship beyond the training interval for which that relationship was initially established, our results would be greatly strengthened. Paramaterizing the sample size by the amount of training data with the same relationship, but extending the training interval to be over three times as long still yielded statistical convergence. This is exhibited in Figure 4.19 with a regression slope of -0.488, an R^2 value of 0.996, and once again a negligible p-value.

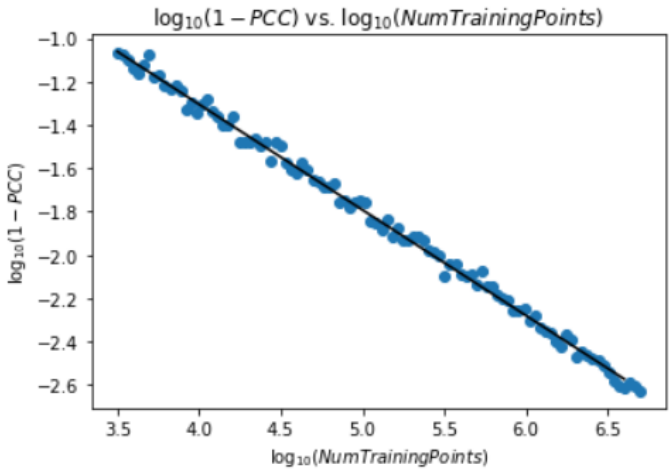


Figure 4.19: Lorenz convergence for varied sample size (II)

For experimental HET signals that were relatively clean and cross-map well, we first assumed that the signals were clean enough to reconstruct each other. Thus, we proceeded as we did with Lorenz test data and added synthetic noise to a signal before attempting to remove it through our denoising algorithm. In particular, the HET Anode+Cathode signal and the Cathode Pearson were prime candidates for this practice, as evidenced in Figure 4.14. For this reason, it made sense to try and optimize the relationship between the sample size and amount of training data used for the HET system. For reasons stated

in Section 4.1, we used the value $\tau = 150$ and an embedding dimension of $d = 8$. We also added Gaussian noise with a standard deviation of 0.25 to the Cathode Pearson signal for each of our trials. Beginning with our usual fixed sample size plot for optimizing the performance of the algorithm, we noticed similar behavior for the HET system, but with less organization than before. Figure 4.20 we depicts the minimal error points, which follow a weakly linear trend, as well as the relationship between the optimal sample size and the amount of training data that was used.

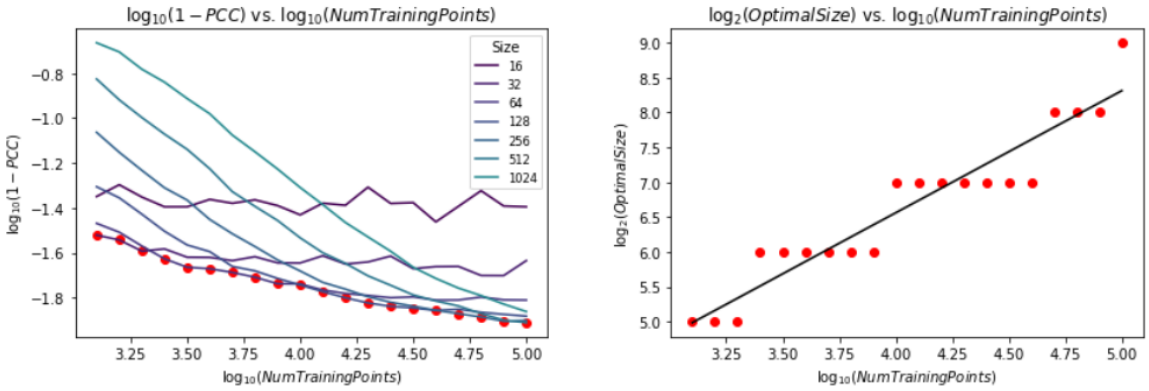


Figure 4.20: Optimizing HET sample size and training data relationship

Using the relationship between the sample size and amount of training data given by the right image in Figure 4.20, we examined the statistical convergence of our Cathode Pearson signal reconstructions. Our findings are exhibited in Figure 4.21 and unfortunately indicate that the method does not seem to be converging completely. We found the regression line to have negligible p-value, an R^2 value of 0.89, and a slope of only -.19. We believe that this lack of convergence could have been caused by the fact that the time series we are using from the HET system have over 100 times less elements than those we are regularly sampling from the Lorenz system. The unpredictability and chaotic nature of the HET also could have contributed to the effect that we saw.

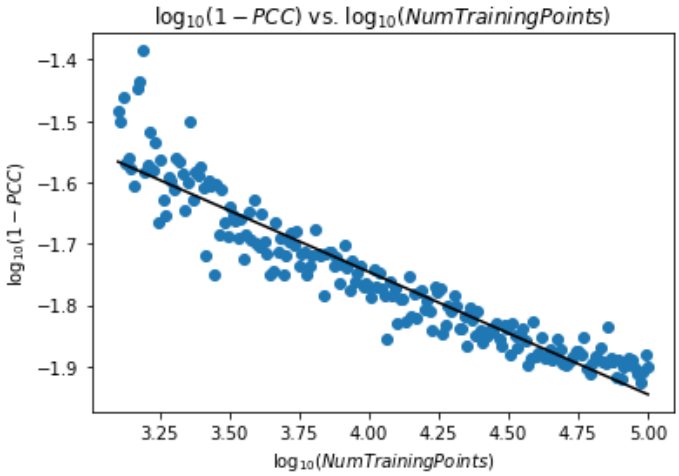


Figure 4.21: Cathode Pearson Convergence Plot

4.3.2 Impact of Noise

We went on to examine how the standard deviation of the applied Gaussian noise affects reconstructions. Shown in Figure 4.22 are convergence plots for a fixed subset size for noise that varies by factors of 2. On the left we reconstructed Lorenz Y in dim 3 with a subset size of 250, while on the right we reconstructed Lorenz Z in dim 3 with a subset size of 300.

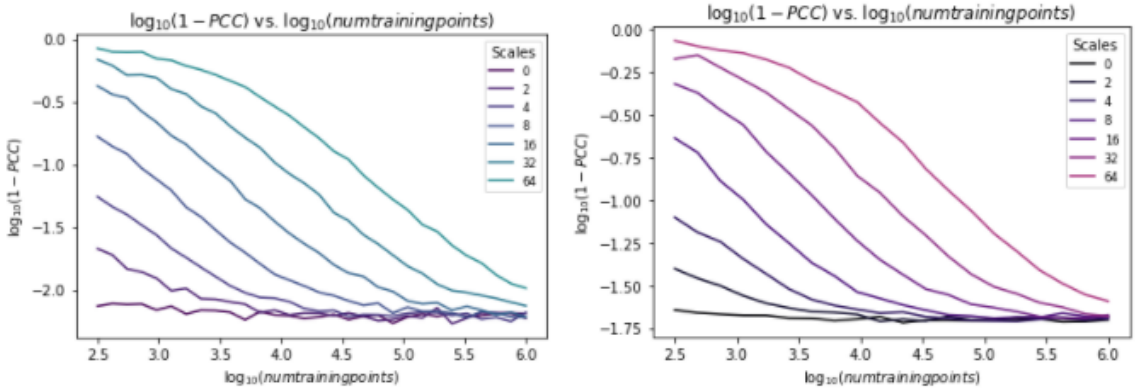


Figure 4.22: Mesh comparison for optimal uniform mesh parameters

We found it interesting that the convergence line for 0 noise was flat and that the rest of the lines seemed to approach that value in the limit as the number of training points went to infinity. This may have been the case because we were sampling our training data from such a large interval that even a relatively small amount of points chosen would approximate the behavior of the Lorenz system well. This, combined with the fact that there was no need for averaging to get rid of noise, seemed to create a fast upper bound on how successful our reconstructions with 0 noise could be for a fixed subset size.

4.3.3 Convergence to an Unknown Clean Signal

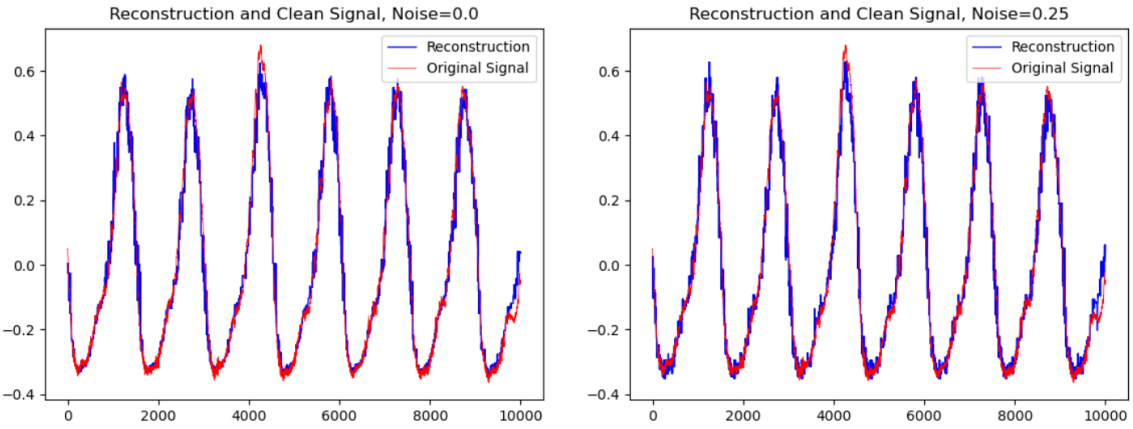


Figure 4.23: Left: No Noise Added, Right: Noise with scale 0.25 Added

In the case where we applied our method to noisy experimental data, we had no truly clean signal to reference for error quantification. Because real data no longer offers us a "true" clean signal to compare our reconstruction to, we instead checked that the resulting reconstructed signal did not change when we add extra synthetic noise to the original

messy signal. A visual representation of this is offered in Figure 4.23, where the impact of differing added noise levels is seen in a reconstruction of the Cathode+Pearson signal. Both a reconstruction without any added noise and a reconstruction with noise added with a scale of 0.25 are shown, and though there were differences in the reconstructions, the general behavior of the reconstructed signal remained quite similar. Also important to note that both reconstructions are created with a dimension of 5, and all other parameters remain consistent between the two attempts.

It is important to note that the HET system has a potentially infinite number of interacting variables, and as a result, there does not exists a one-to-one mapping between any two observables: see Figure 4.24. This causes our algorithm to only be able to reconstruct features of a signal which shares information with the shadow-manifold-generating signal.

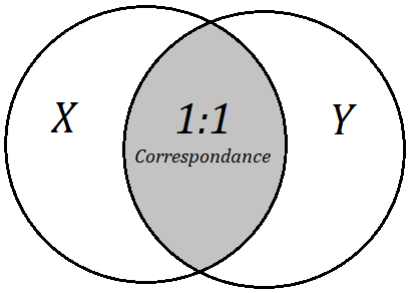


Figure 4.24: Diagram of partial 1-to-1 correspondence between X and Y. The non-shaded information in the Y signal is irretrievable by X.

In Figure 4.25, this issue with an unknown clean signal is illustrated. Both the Ring 1 and Ring 6 data provided create extremely noisy signals. As a result, it is difficult to tell which parts of the signal are noise and which parts contain valuable information about the dynamics of the system. While the reconstructions look good, it is important to note that it is difficult to definitively state how similar the reconstruction is to the true dynamics of the system.

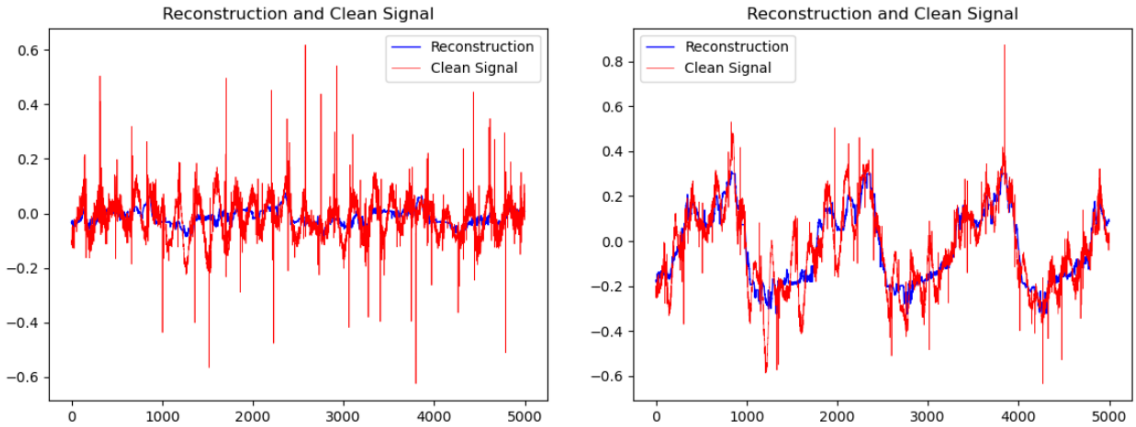


Figure 4.25: Left: Ring 1 Signal Reconstruction, Right: Ring 6 Signal Reconstruction

To quantify convergence in this case, following from Figure 4.23, we considered whether our reconstructions "converged" in the sense that the error between them decreased to 0 as σ went to 0, where σ is the standard deviation of the Gaussian noise added to the original signal. We denote the original signal by Y and the signal with added noise of standard deviation σ by $Y'(\sigma)$.

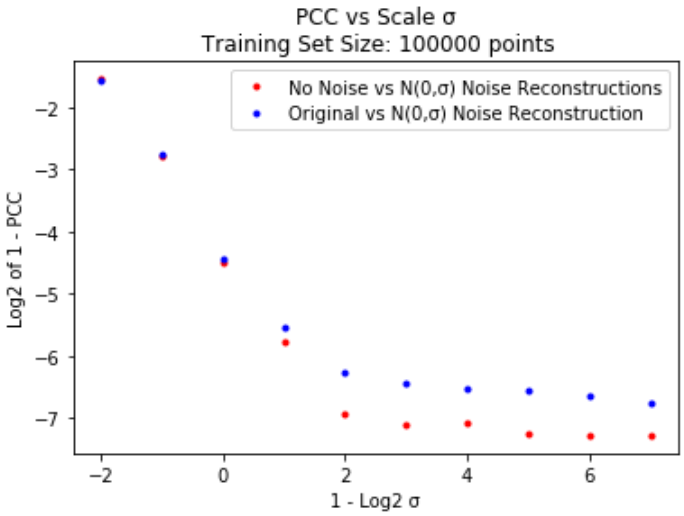


Figure 4.26: A comparison of the error between reconstructions vs between a reconstruction and the original signal

In Figures 4.26, we show in red the error between a reconstruction R_1 built from $Y'(\sigma = 0)$ and a reconstruction R_2 built from $Y'(\sigma)$. In blue, we show the error between the reconstruction R_2 and the original signal Y. We observed that the absolute error between the two reconstructions was less than the error between a reconstruction and the original signal. This could signify that the reconstructions were more similar to each other than the original, suggesting that our algorithm denoised through the noise of the original signal. However, we also observed leveling-off instead of convergence as σ dropped below a threshold. Because of the linearity before the threshold and how the leveling-off occurred rather suddenly, we suspected that this was due to an error in computer calculations. Thus, the reconstructions may have converged after all.

4.4 Locations and causes of error

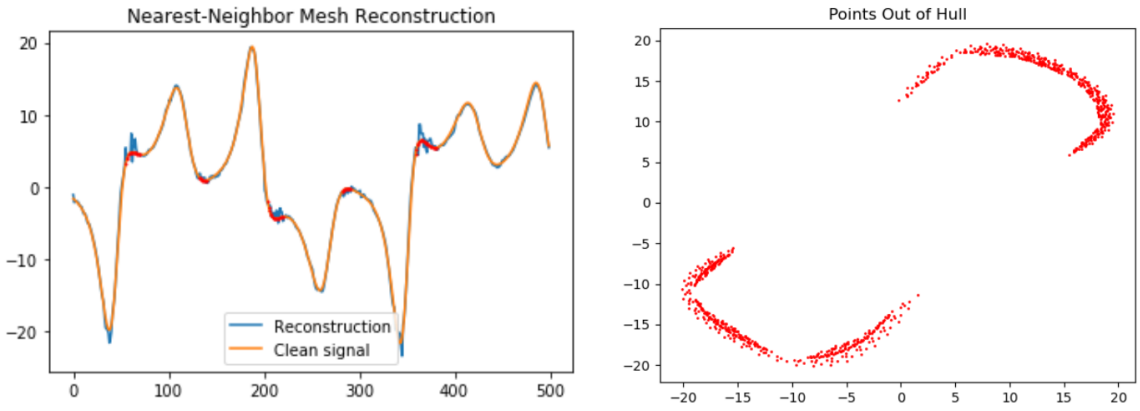


Figure 4.27: Left: The Y Signal with Outliers Labelled, Right: The Outlier Points of the Original System

Finally, we discuss certain limitations of our method which could not be mitigated by

our current refinement and optimization methods, and suggest ways that this might be solved in the future.

Figure 4.27 shows in orange the original signal and in blue the reconstructed signal. We noticed that the majority of visible error in these diagrams were at the local extrema, and furthermore found that many of these extrema with poor reconstructions occur at the times where the the testing points landed outside the convex hull of the cell representatives in the time-delayed embedding. In Figure 4.27, we show these points outside the convex hull on the right, and the Y signal points at these timesteps are displayed in red on the left.

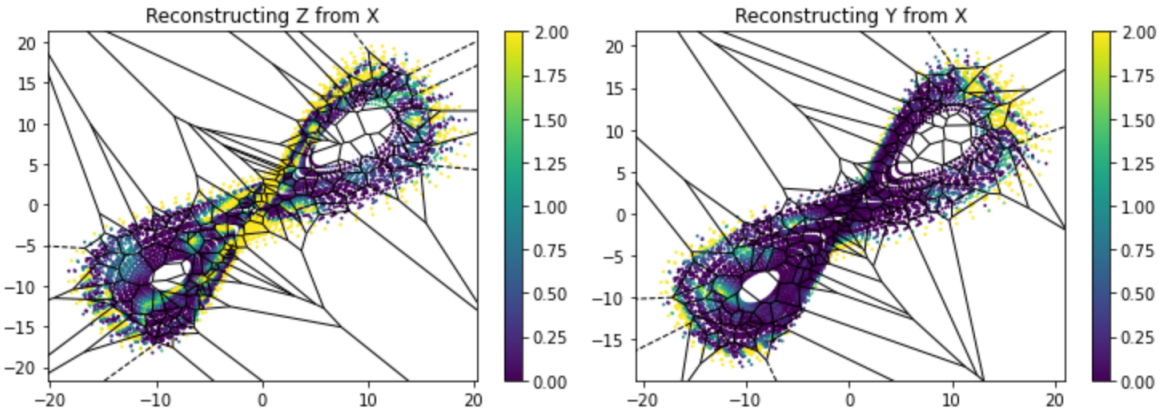


Figure 4.28: Lorenz Error Locations with color

In Figure 4.28, color maps show the areas of the manifold that correspond to the greatest error in the reconstruction. Specifically, the color yellow indicates an area of higher error, while purple indicate an area of lower error. This shows that the highest regions of error in the reconstruction tended to match with outlier and crossing points in the shadow manifold.

Varying values of τ could cause differences in the level of error present in the reconstruction. Often, the additional error at certain values was caused by the presence of crossing in the shadow manifold. This crossing caused points that were further away from each other in the data set to be included in the same cells, which in turn increased the error. In Figure 4.29 depicts similar errors in HET system manifolds, with the color system the same as above Lorenz plots. With these plots, it was clear that all shadow manifolds of the Anode+Cathode data have issues with crossings regardless of τ, as the crossing arose seen in the yellow areas of higher error. These plots also gave more evidence as to why τ was chosen to be 150 in reconstructions, since the manifold with this τ contained the fewest crossing regions.

4.5 Comparison Between Uniform and Unstructured Mesh

Our motivation in this project was to develop a better denoising method than AFRL's previously existing one. Thus, it was important that we investigated the differences in the performance of the uniform mesh method and the unstructured mesh method.

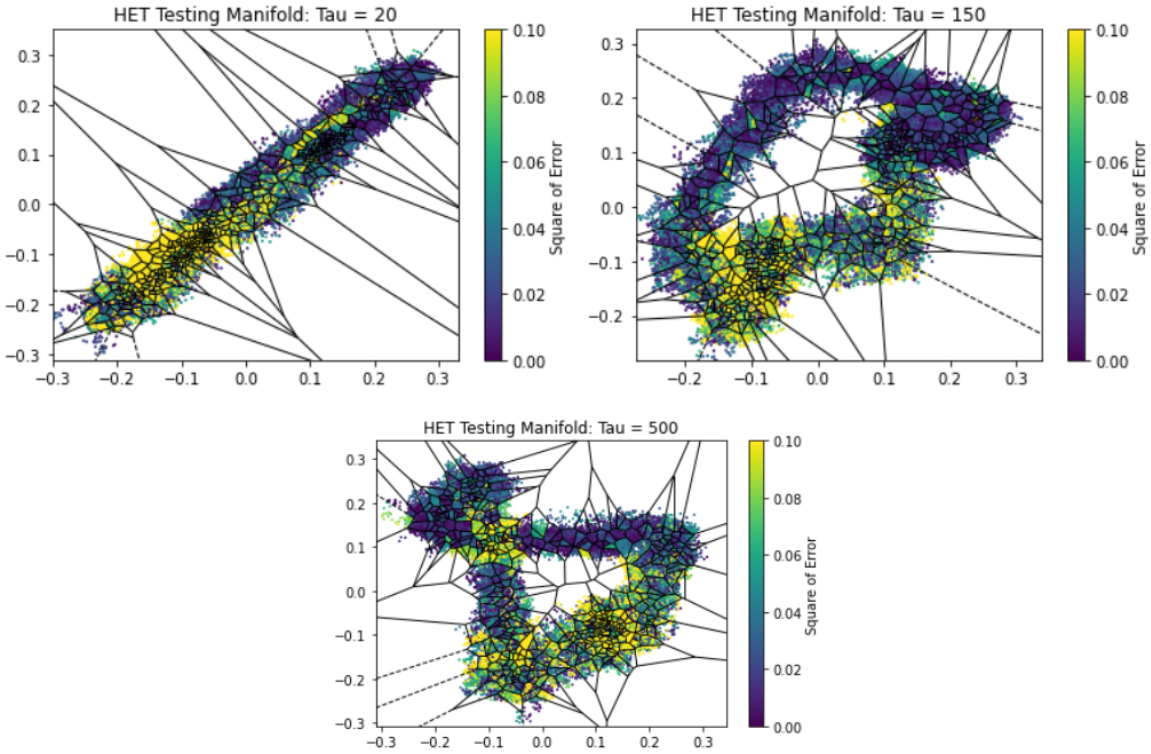


Figure 4.29: Error Color Maps, HET Data

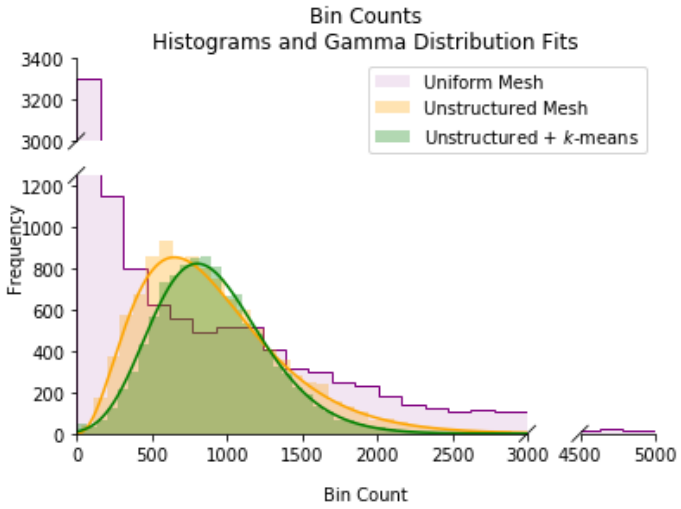


Figure 4.30: A comparison between histograms of the number of points per cell of the uniform mesh, the unstructured mesh, and the unstructures mesh with k-means cell adaptation.

4.5.1 Cell Count Distribution

We initially expected that the bin counts, the number of training points in each cell, would fit a Poisson distribution. However, in Figure 4.30, we see that the Gamma distribution fit the data fairly well. It is also possible that another heavy-tailed distribution, like the lognormal distribution, would provide a good empirical fit. We found that neither the number of training points nor the dimension of the shadow manifold seemed to affect the distribution of bins. We also see that for the uniform mesh, there were many cells with only a few points. This caused the uniform mesh to overweight the values of outlying points.

This effect was especially pronounced when using linear interpolation on the uniform mesh.

4.5.2 Comparison of Convergence

We also sought to compare the convergence properties of the uniform mesh with those of the unstructured mesh. To better understand the relationship between the two methods, we first conducted a comparison biased towards the uniform mesh to observe if the unstructured mesh performed better than the uniform mesh for parameters that it was not optimized with. We then conducted a fair comparison between both methods at their peak performance. It is important to note that we were unable to implement a successful interpolation function on the uniform mesh, so throughout this section the uniform mesh reconstructions are performed without interpolation. For these comparisons to be possible, we had to first optimize the relationship between the number of nonempty cells and the amount of available training data for the uniform mesh.

To do this, we followed an analogous approach to the methods of Section 4.3.1 with slight adaptations for the uniform mesh, for the Lorenz system reconstructing Z from X in dimension 3. We began by plotting error curves for a fixed value of ε in Figure 4.31 and observed very similar behavior to our initial fixed sample size plots for the unstructured mesh.

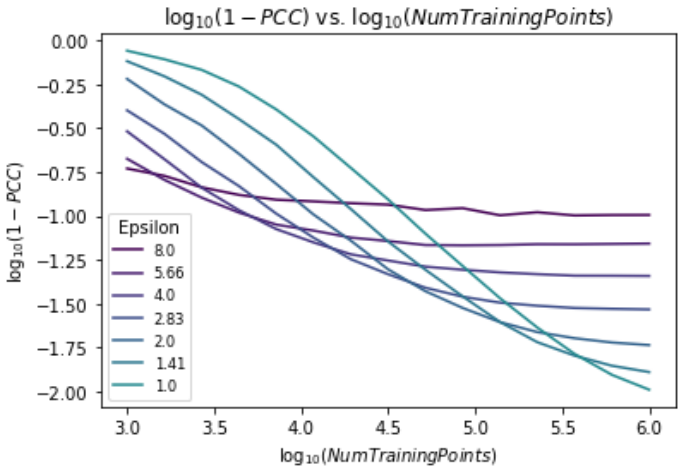


Figure 4.31: Fixed resolution error curves for uniform mesh

Previously, we made sure that the sample size increased by a constant multiple for every test that we ran. For the uniform mesh, the best that we could do is require that the number of cells increased by constant multiples for every value of ε that was tested. As such, the number of nonempty cells would also increase roughly by constant multiples. Therefore, it was sufficient for our purposes to pick a constant $K > 1$ such that every value of ε we test was of the form ε_0/K^n for some positive integer n, where ε_0 was chosen to be the test case with the fewest cells. For this example, we chose $K = \sqrt{2}$, which corresponded to an increase in the number of cells by a factor of roughly 1.6 for each new value of ε that we tested.

Noticing the clear linear trend in optimal reconstruction error, we once again identified the minimal error points and found a linear relationship between the log of ε and the amount of available training data. Figures 4.32 and 4.33 explore this relationship and indicated that

we could optimally vary the value of ε by

$$\log_2 \varepsilon = 5.896 - 0.992 \log_{10}(Num)$$

where Num is the number of data points used for training.

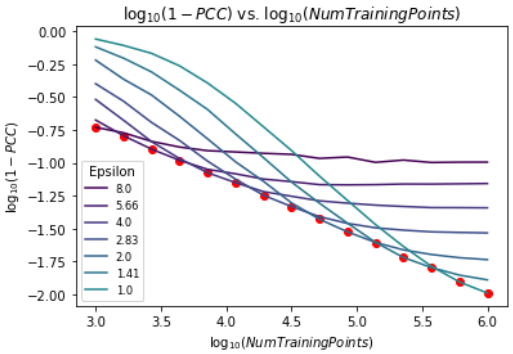


Figure 4.32: Optimal ε values on fixed resolution error curves for uniform mesh

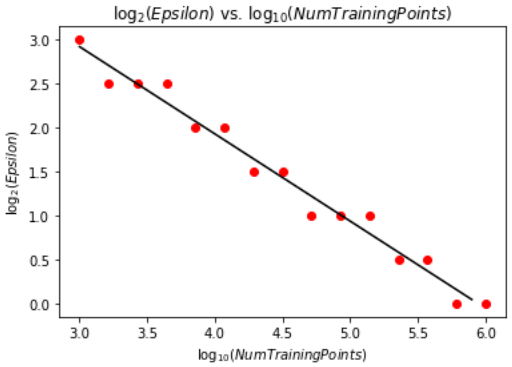


Figure 4.33: Optimal relationship between ε and the amount of available training data

In order to compare the unstructured mesh approach to the uniform mesh while optimally varying ε, we used in each trial exactly the number of cell representatives as there were nonempty cells in the uniform mesh. We found that both methods appeared to converge, but that even when the parameters were chosen optimally for the uniform mesh, the unstructured mesh necessarily had a lower absolute error for a given amount of training data. In Figure 4.34, we show that the unstructured mesh achieved a steeper slope (-0.445) on the left plot than the structured mesh (-0.415). Furthermore, by adding linear interpolation to the unstructured mesh, we could achieve a slope of -0.497.

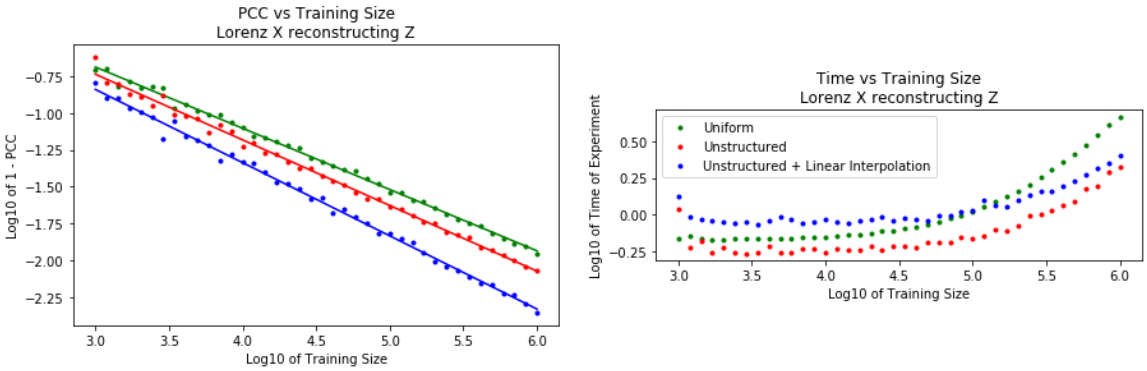


Figure 4.34: Convergence and runtime comparison between two meshes with optimal uniform mesh parameters

The structured mesh ε and the unstructured mesh subset size values were chosen as predicted by the optimization curves as already discussed. Note, however, that we did not optimize the choice of subset size for the unstructured mesh with interpolation curve, so a steeper slope could be achievable. Note that the curves in the right hand plot should not be taken too seriously, as the runtime depends greatly upon the choice of implementation of each part of the algorithm. By changing just one subroutine used in computation, we found that the unstructured mesh computation took longer than the uniform mesh.

Lastly, we also note that computations using k-means clustering to produce the sample points in addition to the linear interpolation produced a curve very similar to the curve with only linear interpolation. In the same vein, using a slightly less-than-optimal curve for computing the subset size given the training size produced a curve very similar to the one for the unstructured mesh. This shows that our method is rather resilient with respect to the function for increasing subset size.

Chapter 5

Conclusions

5.1 Summary

We have accomplished nearly all of our original goals regarding creating and testing a denoising method. We successfully developed a dual unstructured meshing method based on an unstructured mesh using the Voronoi diagram, implemented the refinement strategies of k-means clustering and triangulation-based interpolation, and thoroughly examined the affect of parameters on our reconstruction. Our denoising method successfully reconstructed test data with synthetic noise from the Lorenz, Rossler, and Chen attractors, with success measured based on the error between our reconstruction of a corrupted signal and the original clean signal. We also applied our method to experimental HET data and found strong evidence that our reconstructions are valid even on originally noisy data.

In addition to the development of this method and the production of reconstructions, we developed various methods to test our results and compare them to AFRL's previous uniform mesh method. In particular, we utilized PCC and visual ways to quantify error, and attempted to see how various parameters can impact that error. Through this exploration, we observed the following:

- With a large enough ODE solver timestep or not enough training data, the reconstruction fails to converge.
- Based on the dimensions that we examined, there is an optimal embedding dimension for each type of data.
- The optimal number of cells (sample subset size) is roughly related to the square root of the training time by a constant factor, though with greater training time, this relationship loses importance, as the error of the reconstruction decreases across the board. The impact of different steady subset sizes was also explored, and for the HET data some specific sizes work fairly well.
- The greatest sources of error are the training points which lie outside the convex hull of the Voronoi cell representatives and the crossing regions of shadow manifolds.
- For HET data, crossing regions tend to be the places causing the greatest level of error in reconstruction.
- The distribution of the number of points per Voronoi cell fits a Gamma distribution fairly well for the Lorenz system.

- An optimized application of the unstructured mesh tends to perform better than the uniform mesh.
- Statistical convergence is observed in the optimized unstructured mesh for Lorenz data. This is an extremely promising result to support the success of our method.
- Visually good reconstructions were found for the HET data, especially for sets of data with strong relationships to each other. For the fixed subset size plots on HET data, a trend suggesting convergence with more data was seen (though this is uncertain), which is promising for future tests.

5.2 Future Work

These results are encouraging and suggest further directions for further explorations of this method.

One specific aim for future work on this problem is to test the current method with an input of two noisy signals. Currently, the reconstruction method uses one clean signal and one noisy signal. Clearly, with real data there may not always be a significantly cleaner signal to use in this process. Therefore, making changes to the denoising algorithm that allow it to accept and use two noisy signals to simultaneously denoise each other would increase the robustness of the method and aid in its ability to reconstruct real world information. An unsuccessful attempt at this idea has been made in the past, but we expect that with the use of our method of denoising, another effort may be made with greater chances of success.

Another direction would be to use multiple clean signals to recover one noisy signal when one does not suffice. With the current method, there are often parts of the original signal that are not present in reconstructions. This is because though two signals may be partially related, they may only have a portion of data with 1-to-1 correspondence, and thus there exists information in one signal that may not have hope to be recovered by the other. However, the introduction of another signal may provide 1-to-1 correspondence with the noisy signal in areas that were previously irretrievable. This new method may create more accurate reconstructions of noisier data, and is worth further exploration.

Though this method was inspired by HET data, it has been designed to work for any system resembling a smooth chaotic attractor. Thus, it would be interesting to apply this method to other appropriate real-world systems in the future.

Appendix A

Glossary

A.1 Technical Definitions

Barycenter: For our purposes, center of mass of a simplex.

Barycentric interpolation: Generalization of linear interpolation with barycentric coordinates.

Cell representative: A time-delay embedded data point randomly selected from the shadow manifold which represents a single cell of the unstructured mesh.

Convex hull: The smallest convex set that contains a given set of data.

k-Means clustering: For our purposes, a type of cell-adaptation in which cell representatives are the centers of mass of the cells.

Nearest neighbor: For a subset of data A, the point $p \in A$ is j's nearest neighbor if $d(j,p) \leq d(j,q)$ for all $q \in A$, where d is the Euclidean distance.

Shadow manifold: See: time-delay embedding. The manifold generated by the the vector $\mathbf{P}_X(t) := \Big(X(t), X(t+\tau), X(t+2\tau), \ldots, X(t+(d-1)\tau)\Big)$, where X is a time series signal and τ is the time-delay parameter.

Simplex: Generalization of "triangle" shape in higher dimensions.

Subset: For our purposes, the set of all cell representatives. **Subset size** refers to the number of mesh cells in the Voronoi diagram.

Testing data: The data used from a clean signal in the form of time-delay embedded points used for signal reconstruction in the testing phase of the unstructured mesh reconstruction algorithm.

Time-delay embedding: See: shadow manifold. The manifold generated by the the vector $\mathbf{P}_X(t) := \Big(X(t), X(t+\tau), X(t+2\tau), \ldots, X(t+(d-1)\tau)\Big)$, where X is a time series signal and τ is the time-delay parameter.

Training data: The data used from both a clean signal and a noisy signal to assign average values to a Voronoi diagram in the training phase of the unstructured mesh reconstruction algorithm.

Triangulation: For our purposes, a triangulation of a set of data is a division of the data's convex hull into non-intersecting simplices that can share common faces where the vertices

of the simplices are chosen from the data set, such that all data points are the vertex of some simplex.

Uniform mesh: A grouping of time-delay embedded points constructed using a grid of uniform spacing on the shadow manifold.

Unstructured mesh: A grouping of time-delay embedded data points constructed using the Voronoi diagram on the shadow manifold.

Voronoi diagram: Diagram which groups points into cells based on nearest-neighbor identification. Often used as a synonym for "unstructured mesh."

A.2 Notation

In the order which they were introduced.

$\mathcal{M}^X$: The time-delay embedding (shadow manifold) generated by the time series $X(t)$ of the state variable X.

τ: The time-delay parameter, a nonnegative real number.

$\tilde{Y}(t)$**:** A time series signal corrupted by noise, where the true, noiseless signal is $Y(t)$.

d**:** Time-delay embedding dimension.

V_i**:** One cell of the Voronoi diagram, identified by the index i.

A_j**:** "Average" value associated with the j-th Voronoi cell, V_j.

$R(t)$**:** Time series reconstruction of the signal $\tilde{Y}(t)$.

$Y'(\sigma)$**:** Originally noisy signal Y with extra synthetic noise of standard deviation σ added.

σ**:** Standard deviation of the Gaussian noise added to an originally noisy signal.

ε**:** Side length of uniform mesh cell.

Appendix B

Abbreviations

AFRL: Air Force Research Laboratory. The industrial sponsor of this project.

AMI: Average Mutual Information. A method we used to determine the optimal time-delay parameter.

CCM: Convergent Cross Mapping. A method developed to detect causality between signals.

HET: Hall Effect Thruster. A type of ion thruster studied by AFRL. This is the system which outputs the data to be analyzed in this project.

IPAM: Institute for Pure and Applied Mathematics. An institute of the National Science Foundation, located at UCLA.

ODE: Ordinary Differential Equation. A differential equation based on the derivatives of functions of the independent variables in the system.

PCC: Pearson Correlation Coefficient. A method used in measuring error.

RIPS: Research in Industrial Projects for Students. A regular summer program at IPAM, in which teams of undergraduate (or fresh graduate) students participate in sponsored team research projects.

UCLA: University of California, Los Angeles.

Selected Bibliography Including Cited Works

[1] D. Eckhardt, K. Koo, R. Martin, M. Holmes, and K. Hara, "Spatiotemporal data fusion and manifold reconstruction in Hall thrusters," *Plasma Sources Science and Technology*, vol. 28, p. 045005, 4 2019.

[2] C.M. Greve, K. Hara, R.S. Martin, D.Q. Eckhardt, and J.W. Koo, "A Data-Driven Approach to Model Calibration or Nonlinear Dynamical Systems," *Physics of Plasmas*, 2019. (Accepted).

[3] T. Gallagher, R. Martin, and V. Sankaran, "Unique Identification of Turbulent Resting System Dynamics with Time-Lab Phase Portraits," in *11th U.S. National Combustion Meeting*, 3 2019.

[4] R. Martin and D. Eckhardt, "Denoising of Quasi-Steady Particle Simulations," *31st International Symposium on Rarefied Gas Dynamics, AIP Conference Proceedings*, 2019

[5] A.S. George, C.E. Chan, G. Dimand, R.M. Chakmak, C. Falcon, D. Eckhardt, and R. Martin, "A method to decompose chaotic signals," *In Preparation*, 2020.

[6] S.J. Araki, J. Koo, R.S. Martin and B. Dankongkakul, "A grid-based nonlinear approach to noise reduction and deconvolution for coupled systems," *Physica D (Submitted)*, 2019.

[7] F. Takens, "Detecting strange attractors in turbulence," in *Dynamical systems and turbulence, Warwick 1980*, pp. 366-381, Springer, 1981.

[8] G. Sugihara, R. May, H. Ye, C.-h Hsieh, E. Deyla, M. Fogarty, and S. Munch, "Detecting causality in complex ecosystems," *Science*, vol. 338, no. 6106, pp. 496-500, 2012.

[9] K. Ogata, *Modern Control Engineering* Instrumentation and controls series, Prentice Hall, 2010.

[10] J. Schoukens and L. Ljung, "Nonlinear system identification: A user-oriented road map," *IEEE Control Systems Magazine*, vol. 39, no. 6, pp. 28-99, 2019.

[11] H. D. I. Abarbanel, R. Brown, J. J. Sidorowich, and L.S. Tsimring, "The analysis of observed chaotic data in physical systems," *Rev. Mod. Phys.*, vol. 65, pp. 1331-1392, Oct 1993.

[12] E. R. Deyle, and G. Sugihara, "Generalized theorems for nonlinear state space reconstruction," *PLoS One*, vol. 6, no. 3, 2011.

[13] A. Krakovska, J. Jakubik, M. Chvostekova, D. Coufal, N. Jajcay, and M. Palus, "Comparison of six methods for the detection of causality in a bivariate time series," *Phys. Rev. E*, 2018.

[14] R. Martin, J. Koo, and D. Eckhardt, "Impact of embedding view on cross mapping convergence," *In Preparation*, 2019.

[15] K. Fukunaga, *Introduction to Statistical Pattern Recognition* Academic Press, 1990.

[16] A. Fraser and H. Swinney, *Independent coordinates for strange attractors from mutual information* Phys. Rev. A, 1986.

[17] L. Cao, *Practical method for determining the minimum embedding dimension of a scalar time series* Nonlinear Phenomena, 1997.

[18] A. Leontitsis, *Mutual Average Information* MATLAB Central File Exchange, 2020.

An Unstructured Mesh Approach to Nonlinear Noise Reduction

Jonah Botvinick-Greenhouse Marianne DeBrito
Aaron Kirtland Megan Osborne

Academic Mentor: Casey Johnson
Industry Mentors: Dr. Robert Martin, Dr. Daniel Eckhardt

Our Team

Presenters:

Marianne DeBrito
(Project Manager)
University of Michigan

Megan Osborne
University of Scranton

Jonah Botvinick-Greenhouse
Amherst College

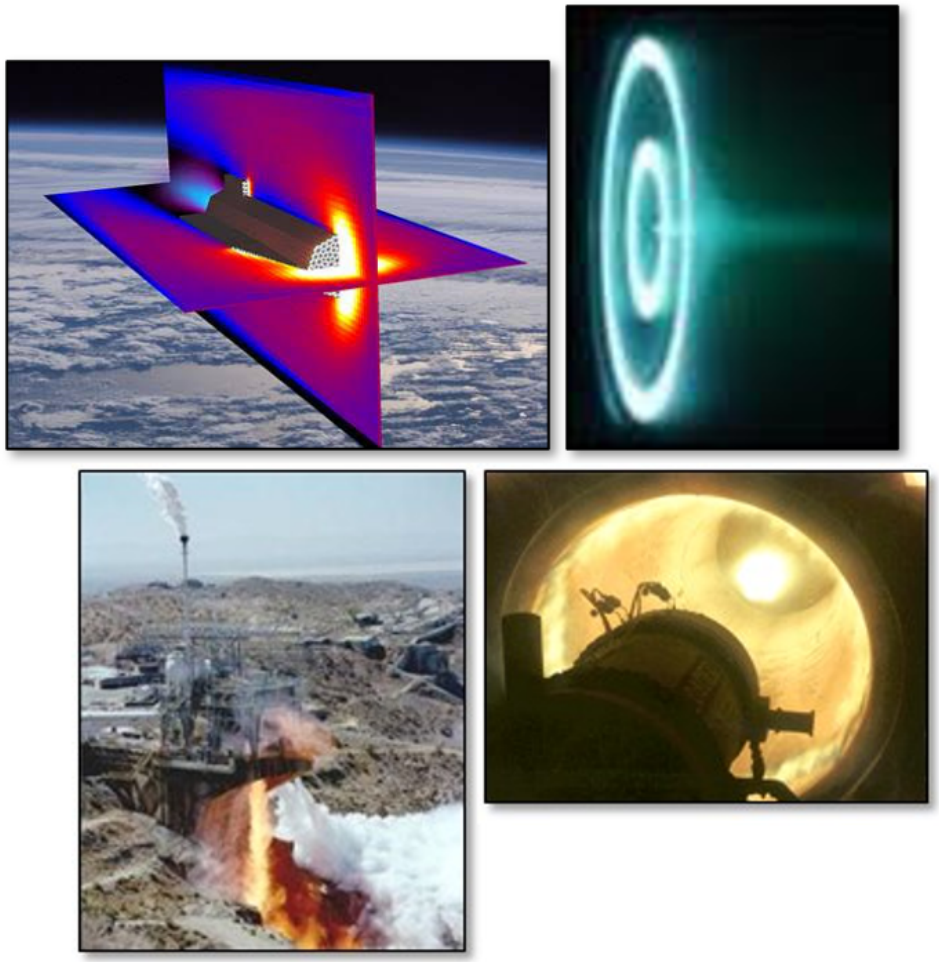
Aaron Kirtland
Washington University in St. Louis

Audience: please mute your microphones.

Air Force Research Laboratory (AFRL)

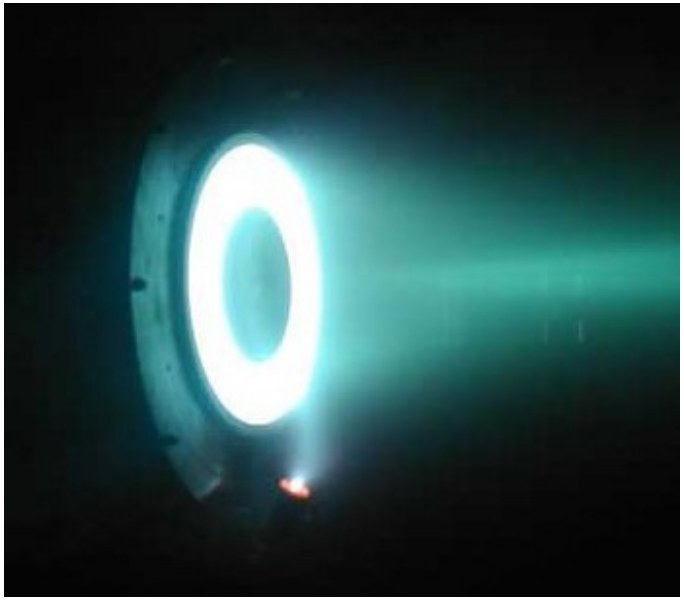
Discover & develop wartime technology for atmospheric and outer-space flight

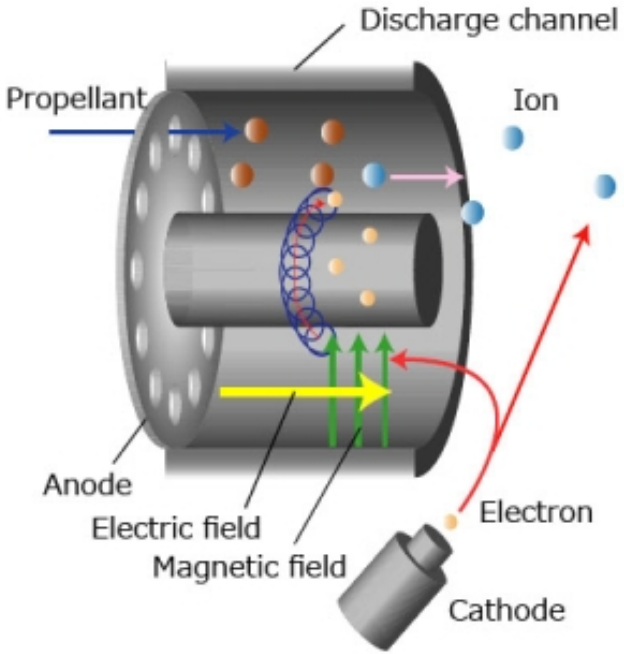
- Aircraft
- Sensors
- Satellites
- Thrusters



⋆ In-Space Propulsion Branch ⋆
Edwards Air Force Base

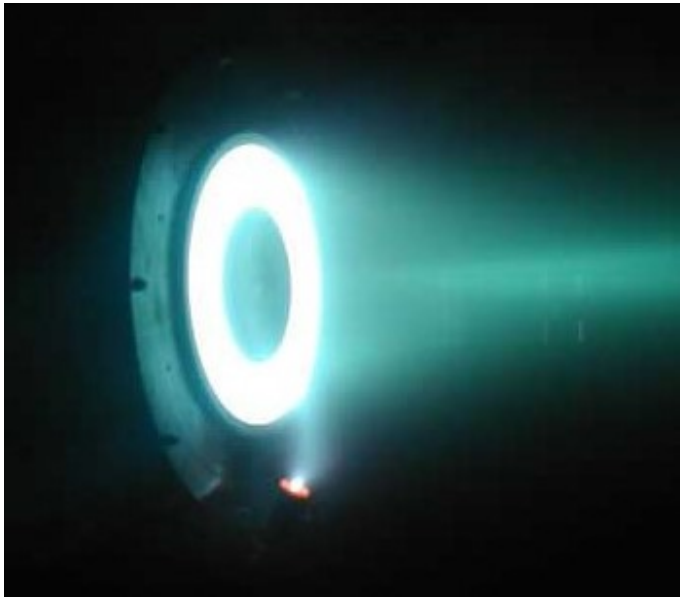
Hall-Effect Thrusters (HETs)

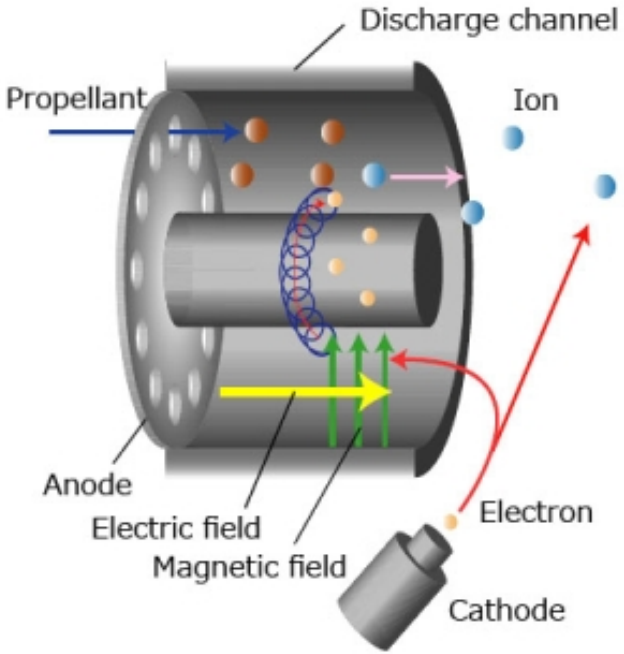




- Ion thruster first developed in Russia
- Studied by AFRL since 90s
- Used on satellites and outer-space robotics
- 100% success rate!
- Dynamical system

Hall-Effect Thrusters (HETs)





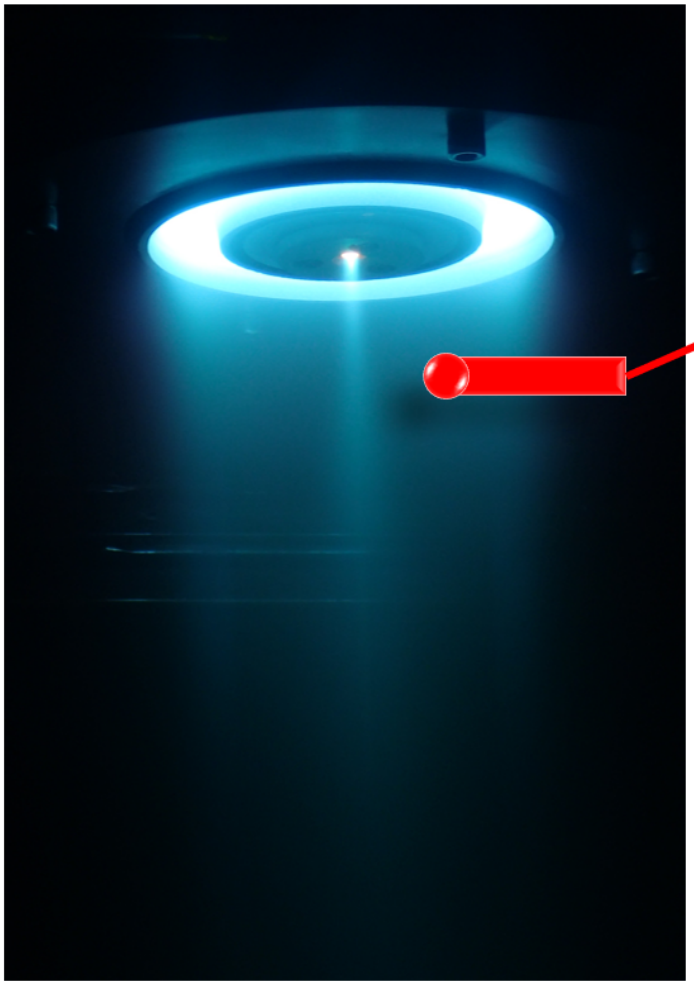
- Ion thruster first developed in Russia
- Studied by AFRL since 90s
- Used on satellites and outer-space robotics
- 100% success rate!
- Dynamical system

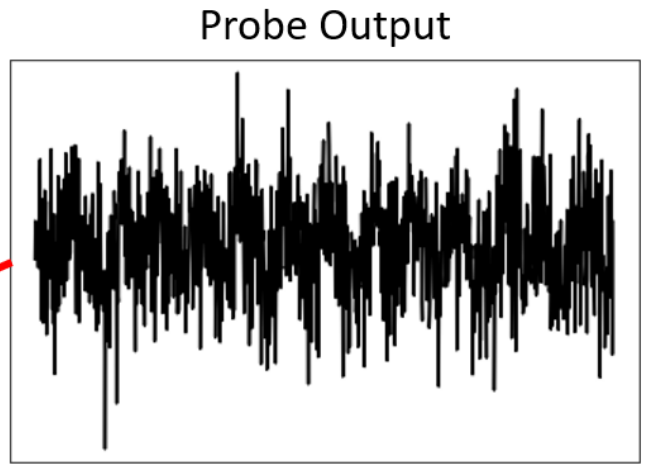
HET Observable Variables

- Cathode current
- Anode current
- Exhaust current (plasma ions)
- Many more

Testing HETs: Noise

Noise corrupts signal, making it hard to see the system's dynamics.





Causes of noise include...

- Electronic acquisition
- Environmental vibrations
- Bigger probes
- Higher resolution

A New Approach to Noise Reduction: Project Goals

Goal: For any smooth chaotic attractor, denoise a corrupted signal using a clean signal sampled from the same system.

A New Approach to Noise Reduction: Project Goals

Goal: For any smooth chaotic attractor, denoise a corrupted signal using a clean signal sampled from the same system.

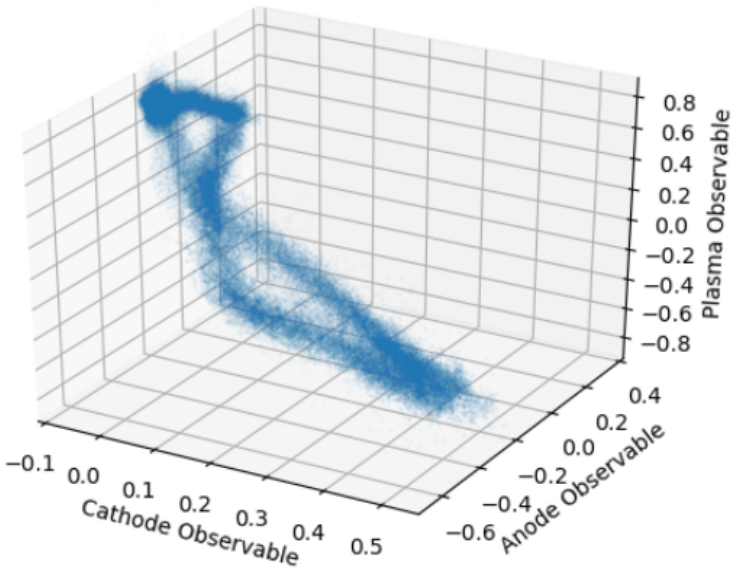
Four phases:

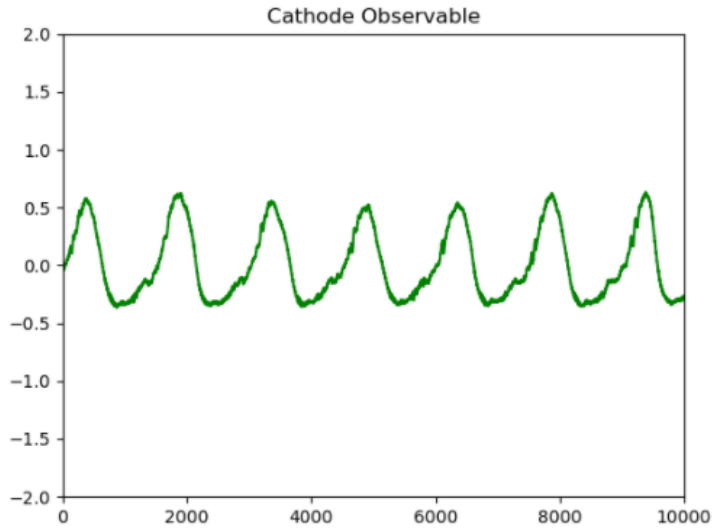
1. **Develop** a new, better algorithm to reduce noise
2. **Test** it on synthetic noise
3. **Analyze** its success and limitations
4. **Apply** it to real HET data

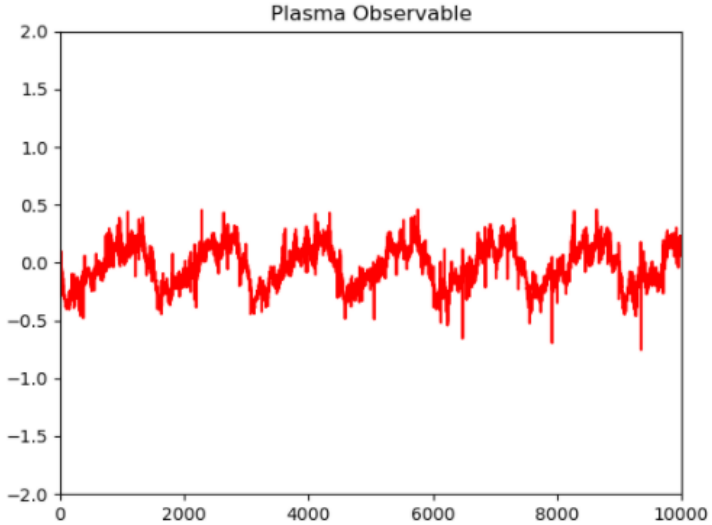
Background

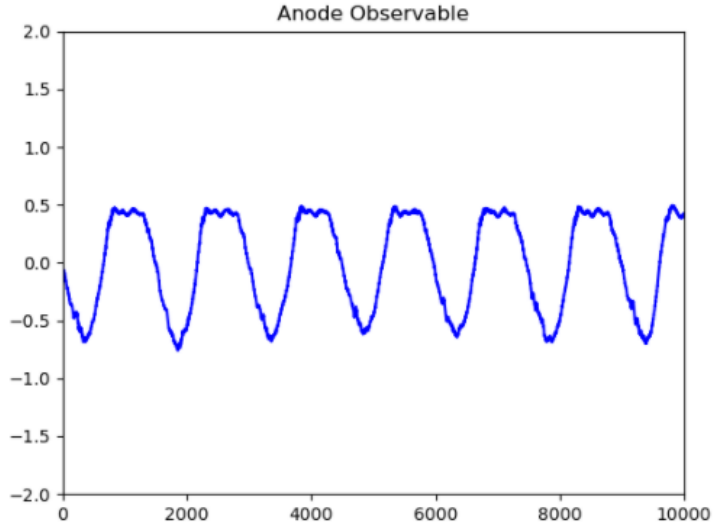
- HET data as a dynamical system
- Time-delay Embedding
- Past attempt: Uniform Mesh

Experimental Data: The HET Dynamical System

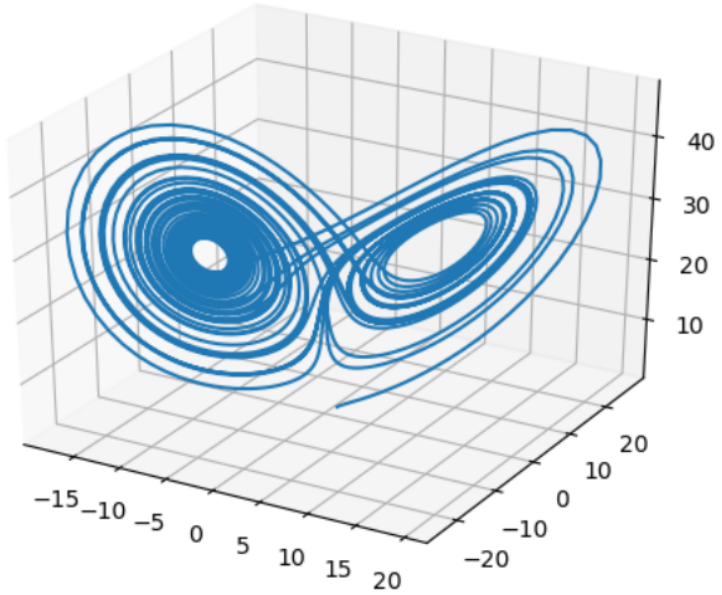


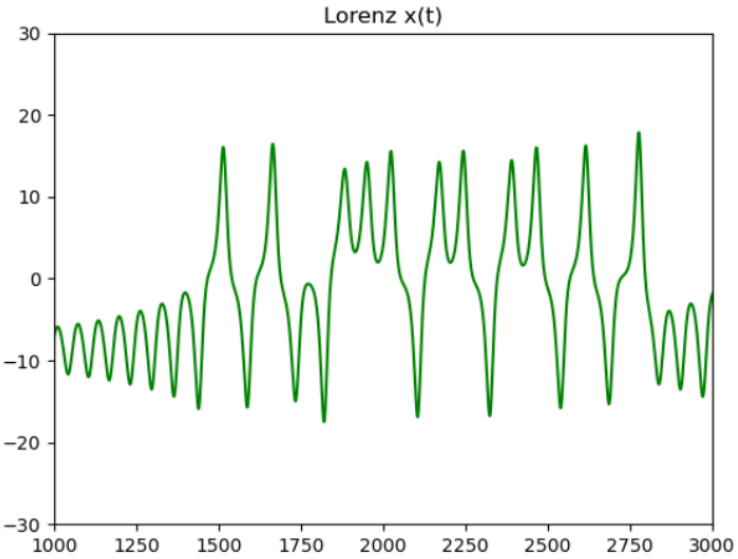


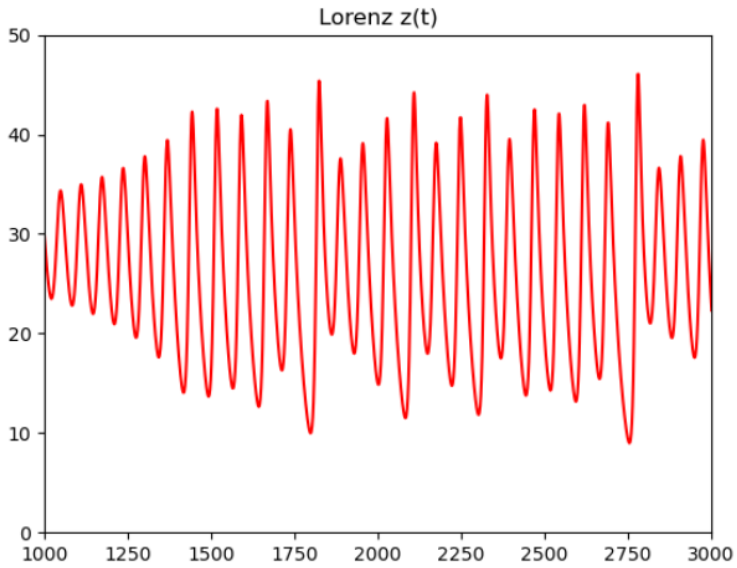


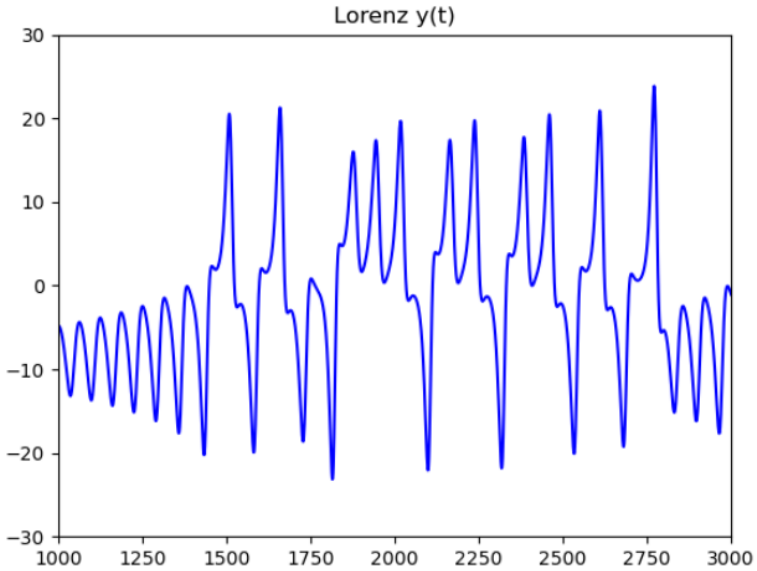


Test Data: Lorenz System







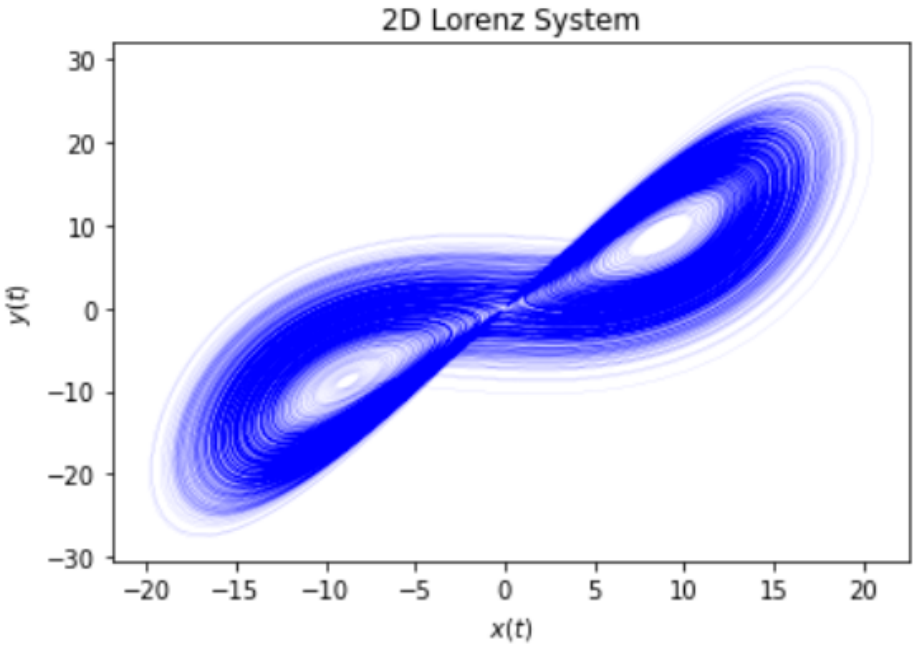


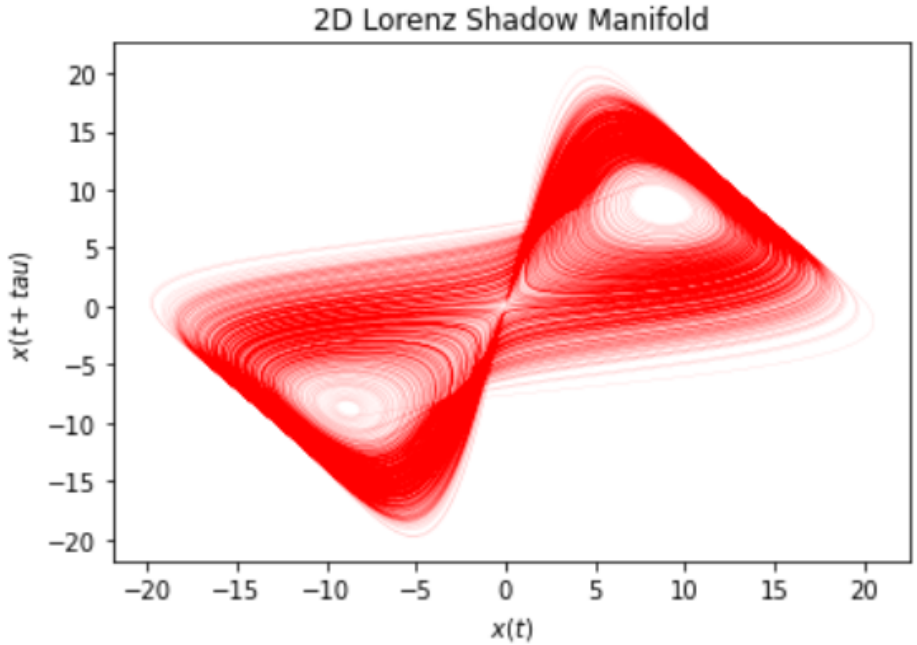
Shadow Manifold: Time-Delay Embedding

Theorem (Summary of Takens' Theorem)

For a high enough time-delay embedding dimension, the embedded shadow manifold recovers the dynamics of the entire system.

- Lorenz axes: $X(t), Y(t)$
- Shadow Manifold axes: $X(t), X(t+\tau)$ for time-delay τ





Lorenz Shadow Manifold Dependence on τ

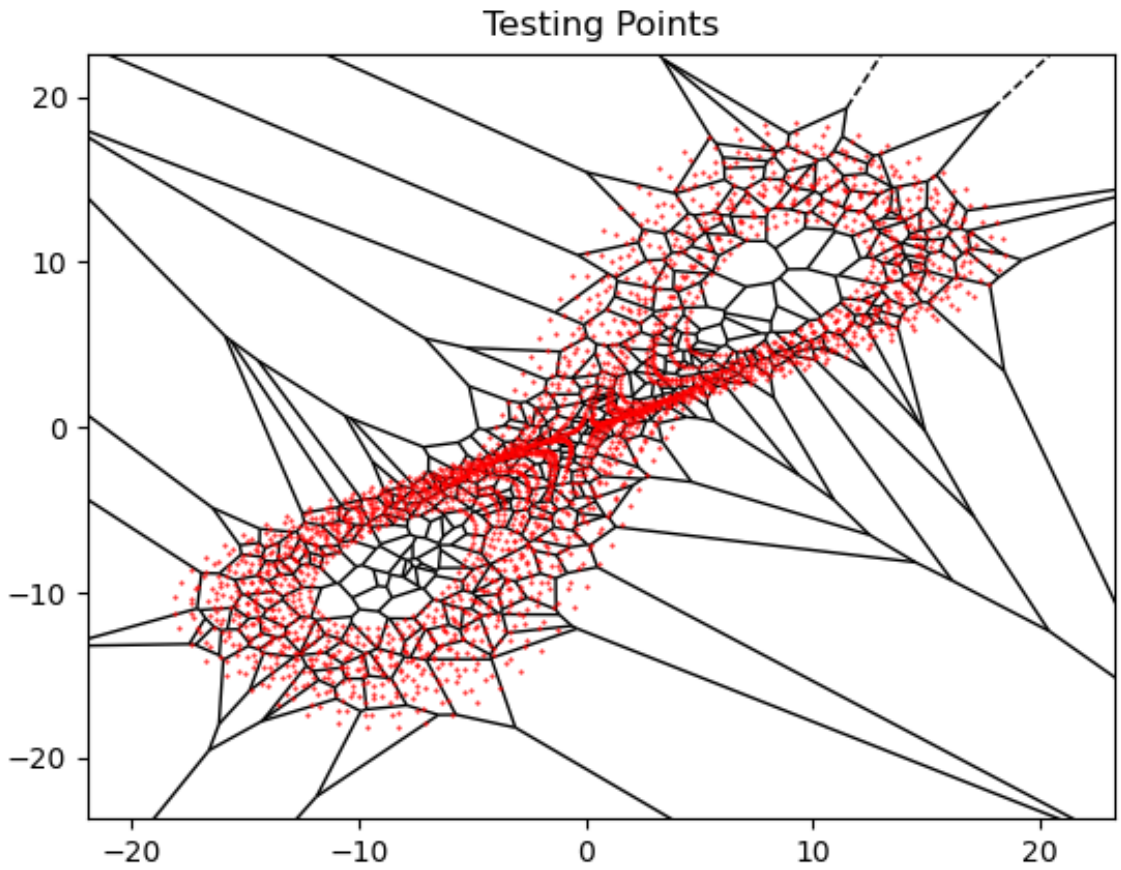


Figure: Shadow manifold dependence on τ

HET Shadow Manifold Dependence on τ

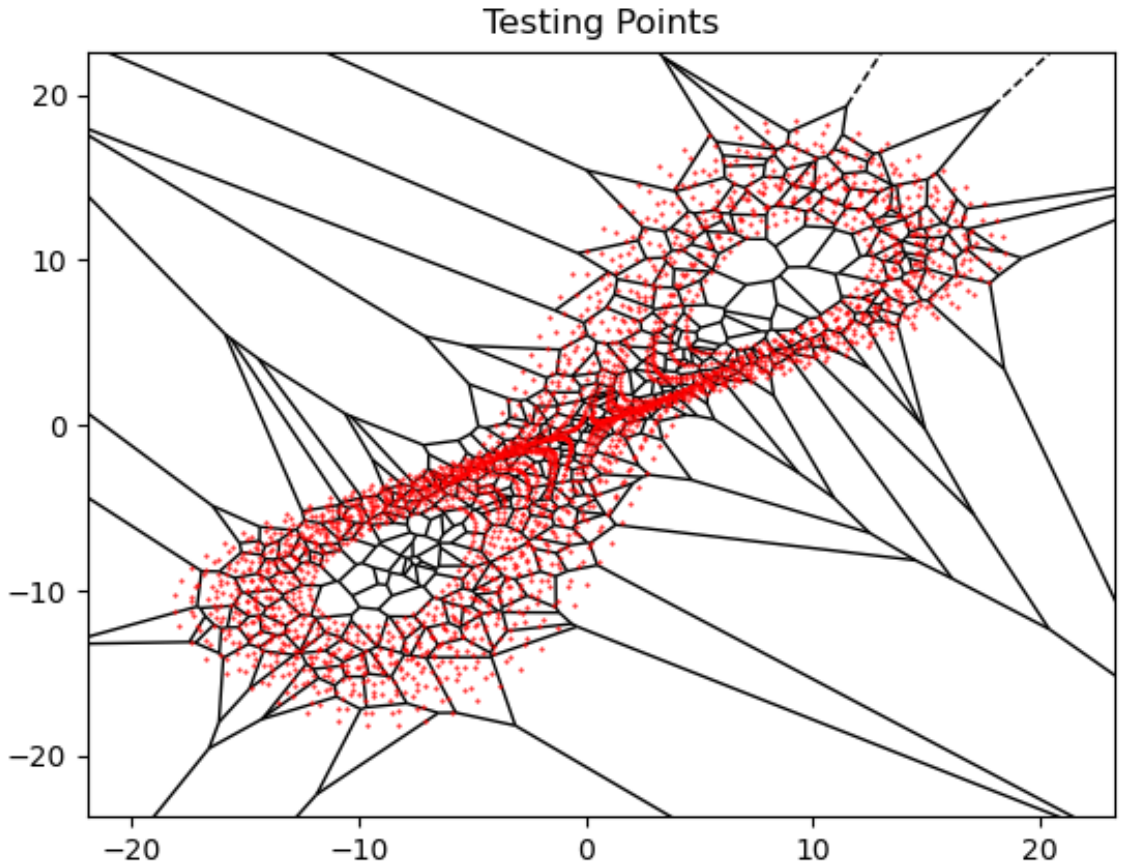
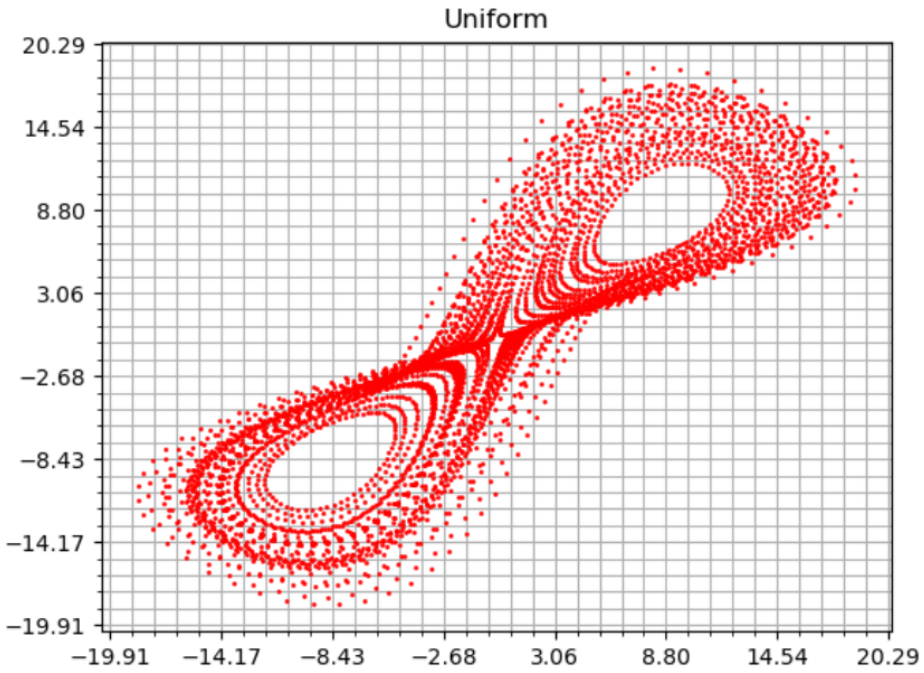
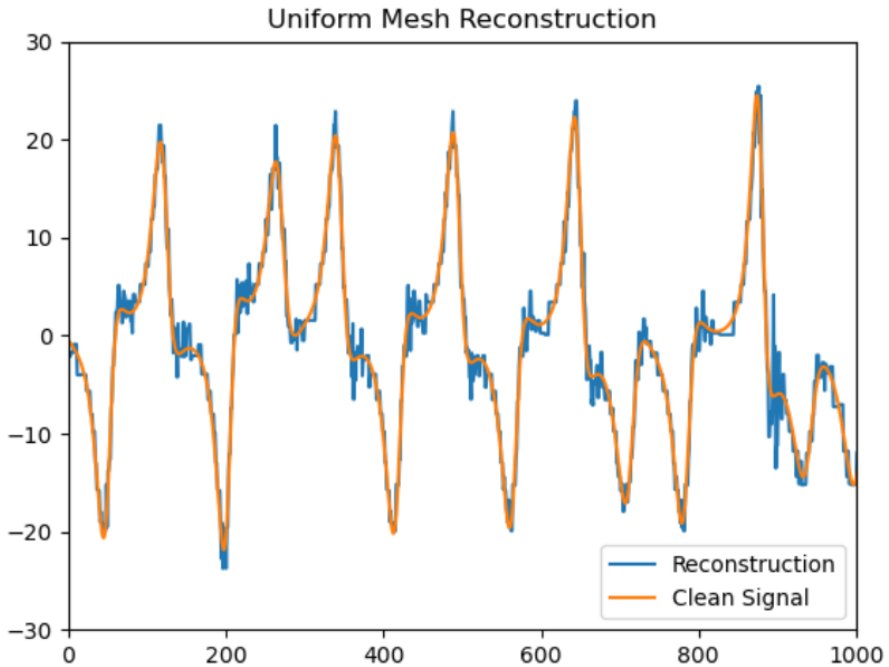


Figure: Shadow manifold dependence on τ

Denoising Process: Uniform Mesh

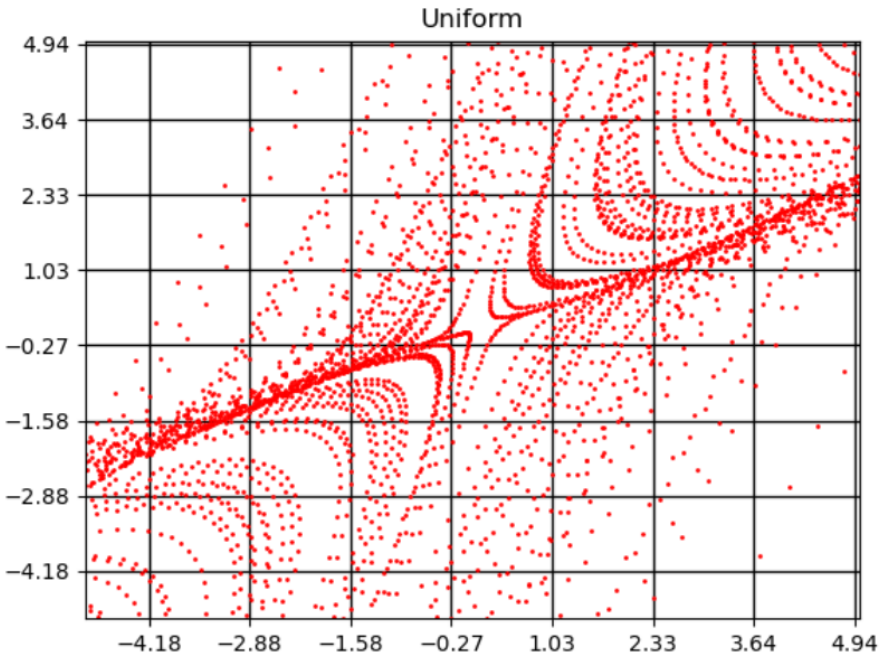
In prior work, AFRL applied a uniform mesh (grid) to Lorenz attractor data in order to recover an original signal.

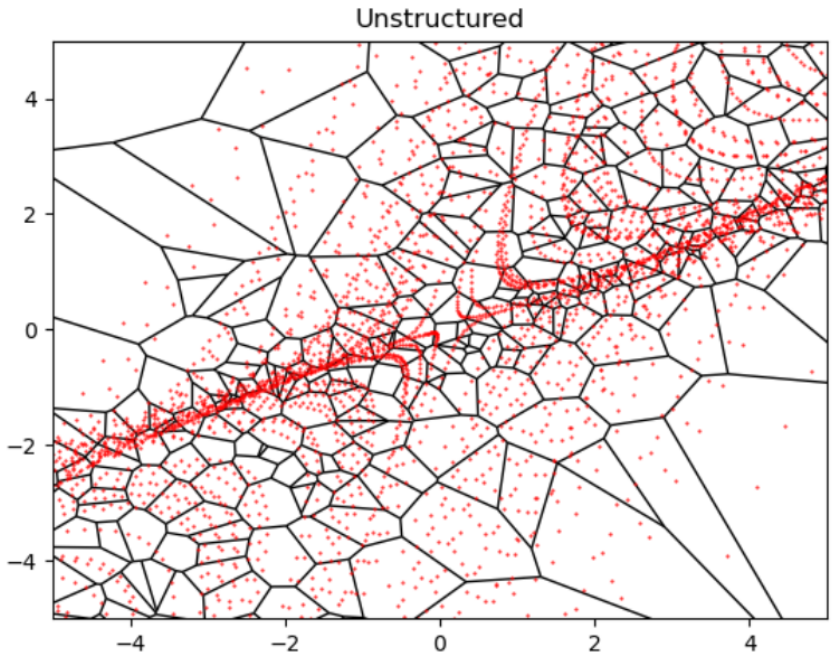




Ditching the Uniform Mesh

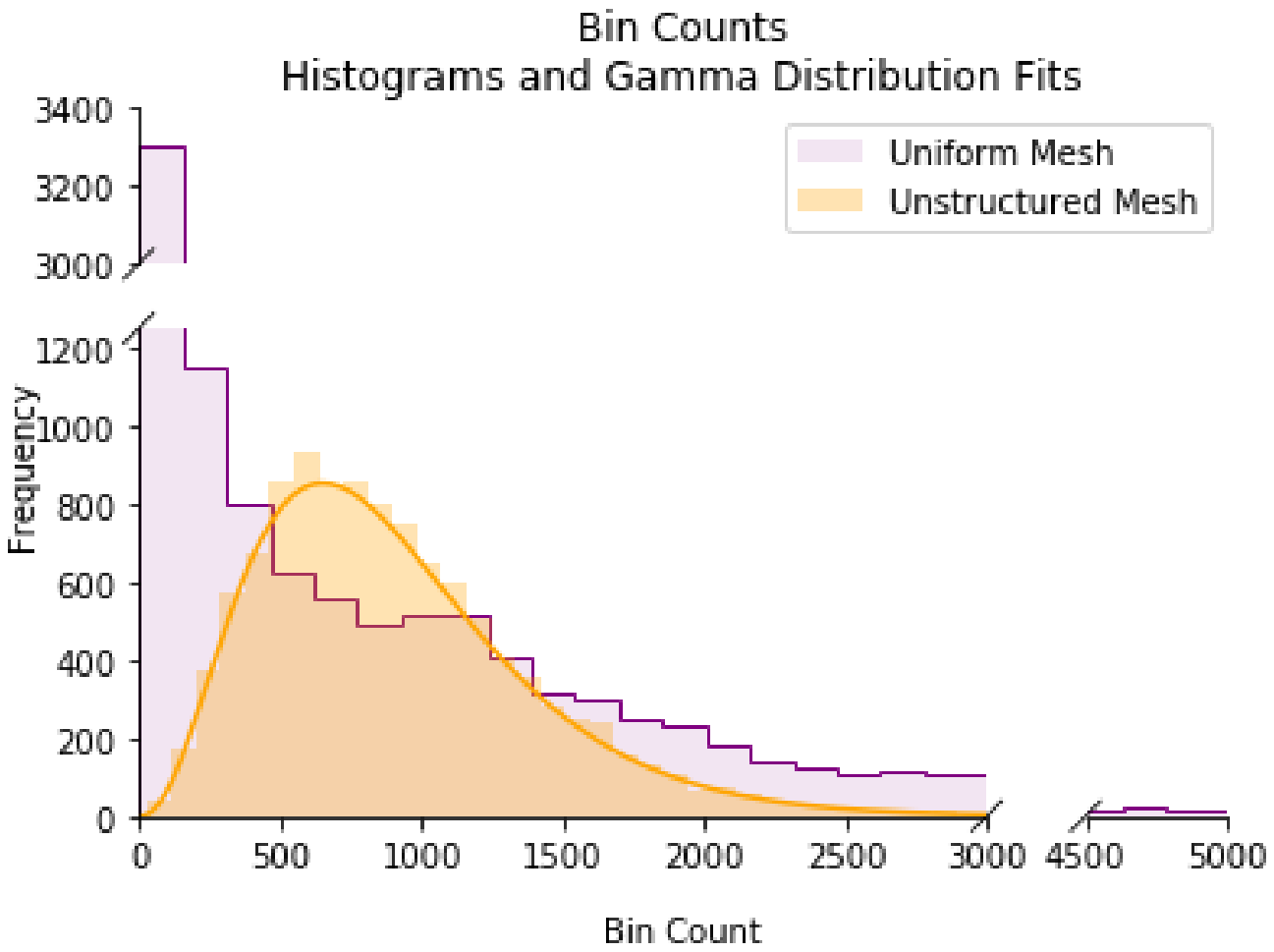
The uniform mesh cells do not adapt to density gradient of data points, so we will consider an unstructured mesh instead.





Ditching the Uniform Mesh

Clearly, the uniform mesh has a significantly higher number of empty or nearly empty cells. The unstructured mesh fits the data better.



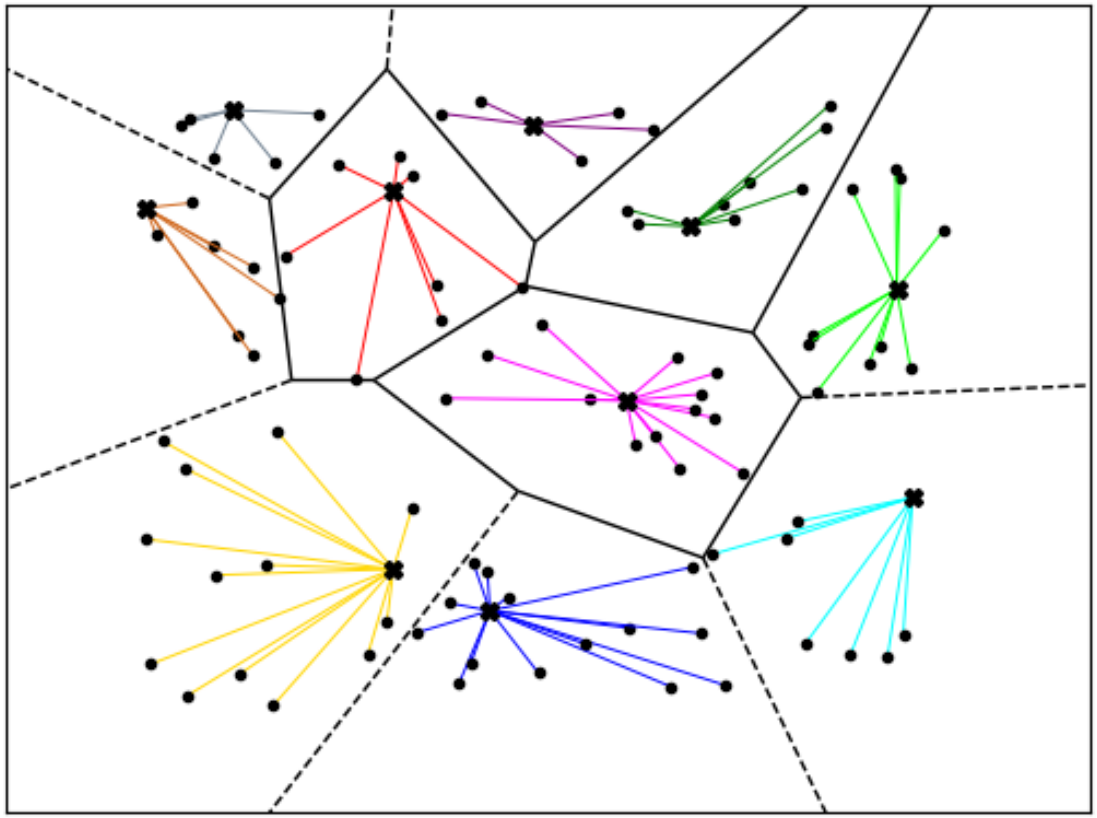
Results: Denoising Algorithm

- Unstructured mesh
- Signal reconstruction

Drawing an Unstructured Mesh: Nearest-Neighbor

1. Randomly select a subset of the points
2. Group points based on the nearest subset points to them

Referred to as the Voronoi diagram



Voronoi Diagram on a Shadow Manifold

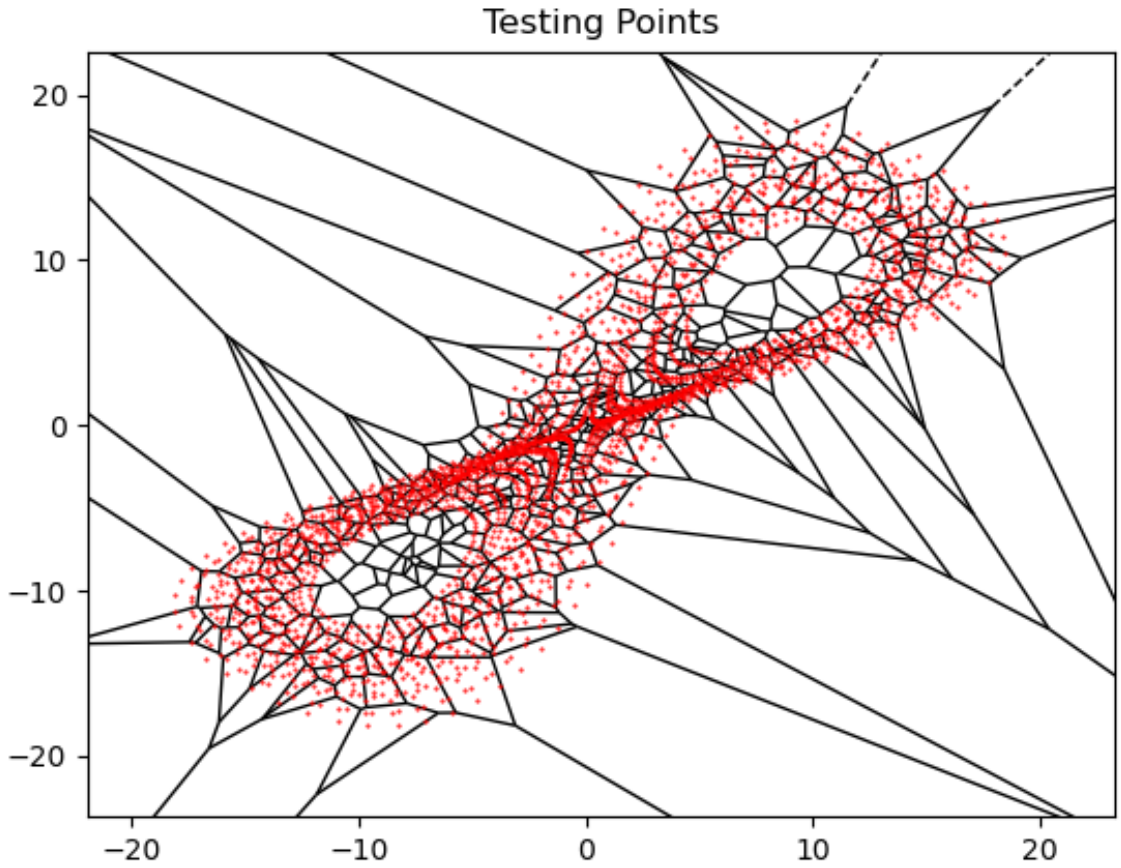


Figure: Movement of unstructured mesh

Reconstruction Method

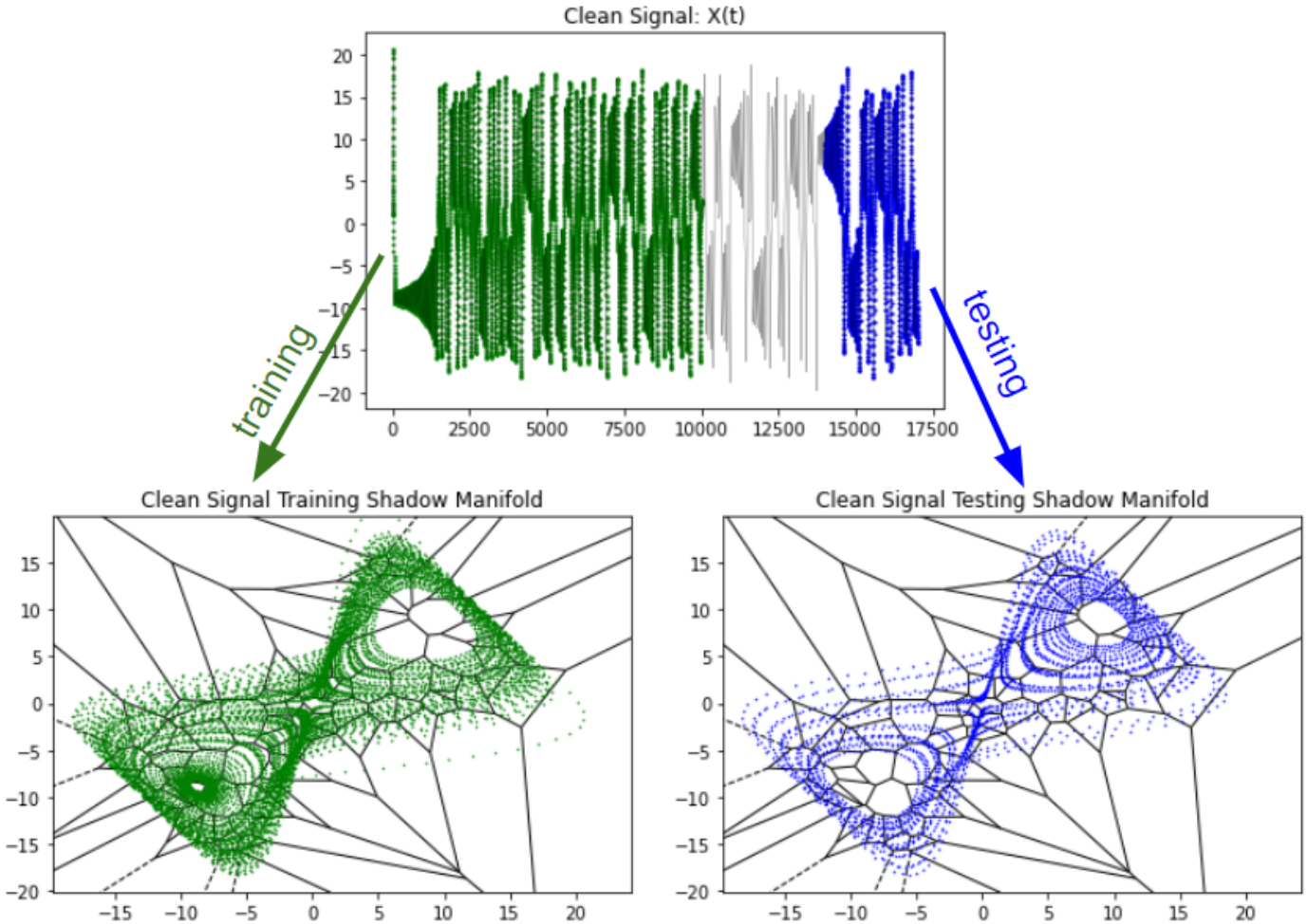
1. **Data Preparation**: Split into training and testing data, and construct shadow manifolds with Voronoi diagrams.

2. **Training**: Average the noisy signal on cells of the clean training shadow manifold.

3. **Testing**: Reconstruct the noisy signal.

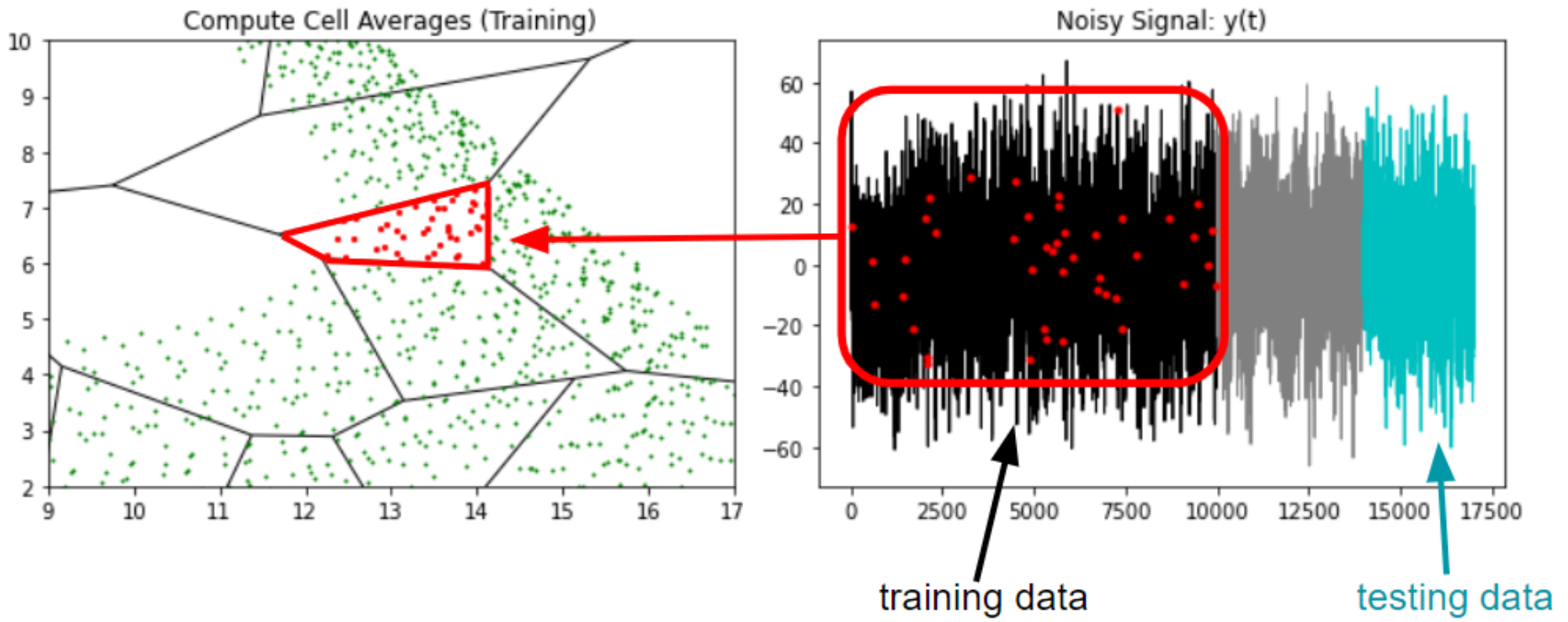
Reconstruction Method (Part 1: Data Preparation)

1. Split the data into training and testing phases.
2. Construct the shadow manifolds and the Voronoi diagram.



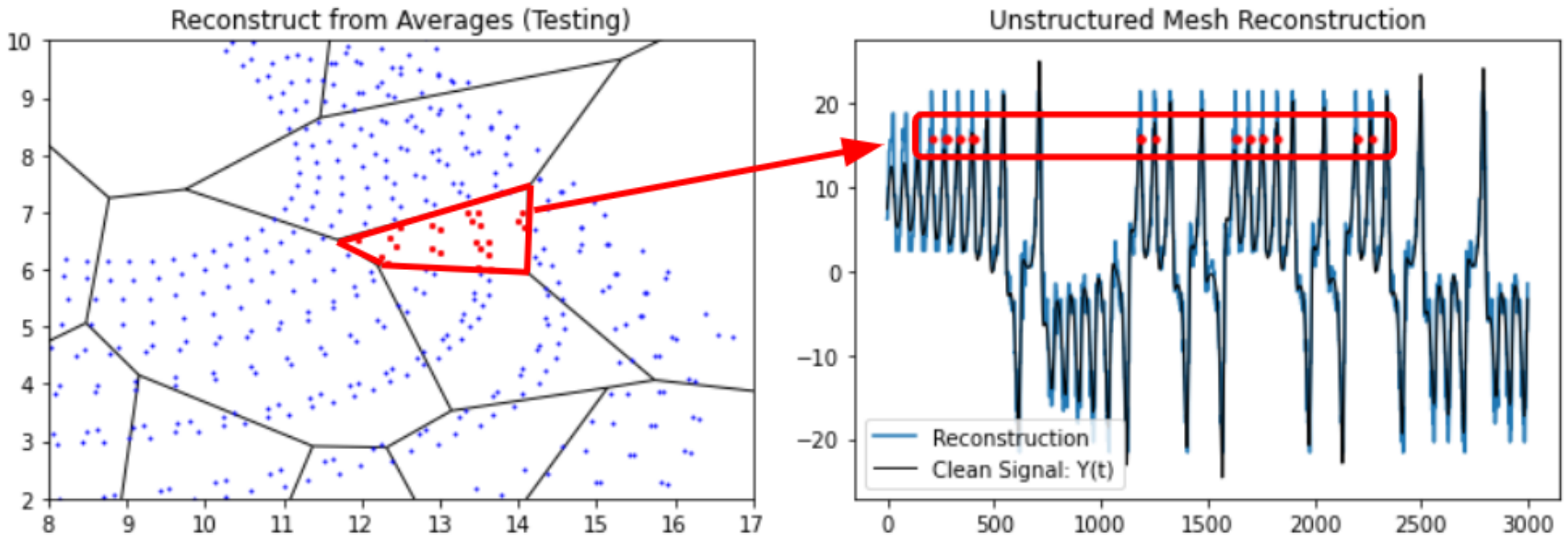
Reconstruction Method (Part 2: Training Phase)

3. Determine the training points of the noisy signal that map to the same cell on the clean training manifold.
4. Compute the average of these values.



Reconstruction Method (Part 3: Testing Phase)

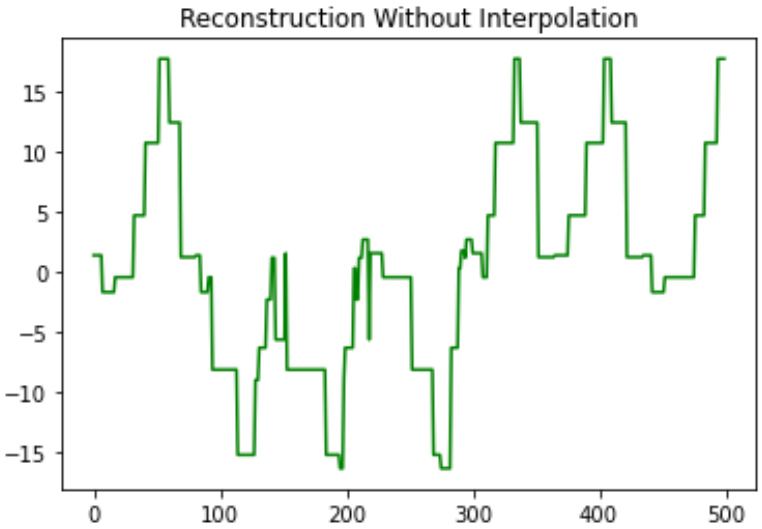
5. Identify the test points of the corrupted signal that map into the same cell.
6. For the time step associated with each of these points, set the value of the reconstruction equal to the cell average.

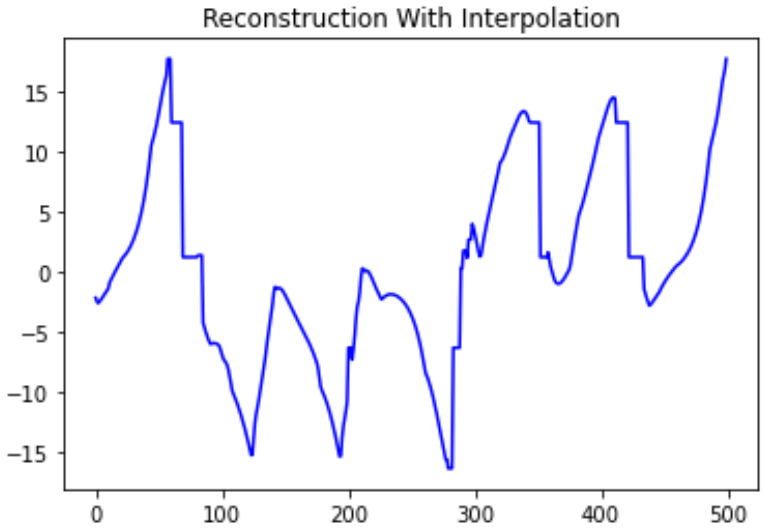


Interpolation

Construct a triangulation of the cell representatives and for all testing points p:

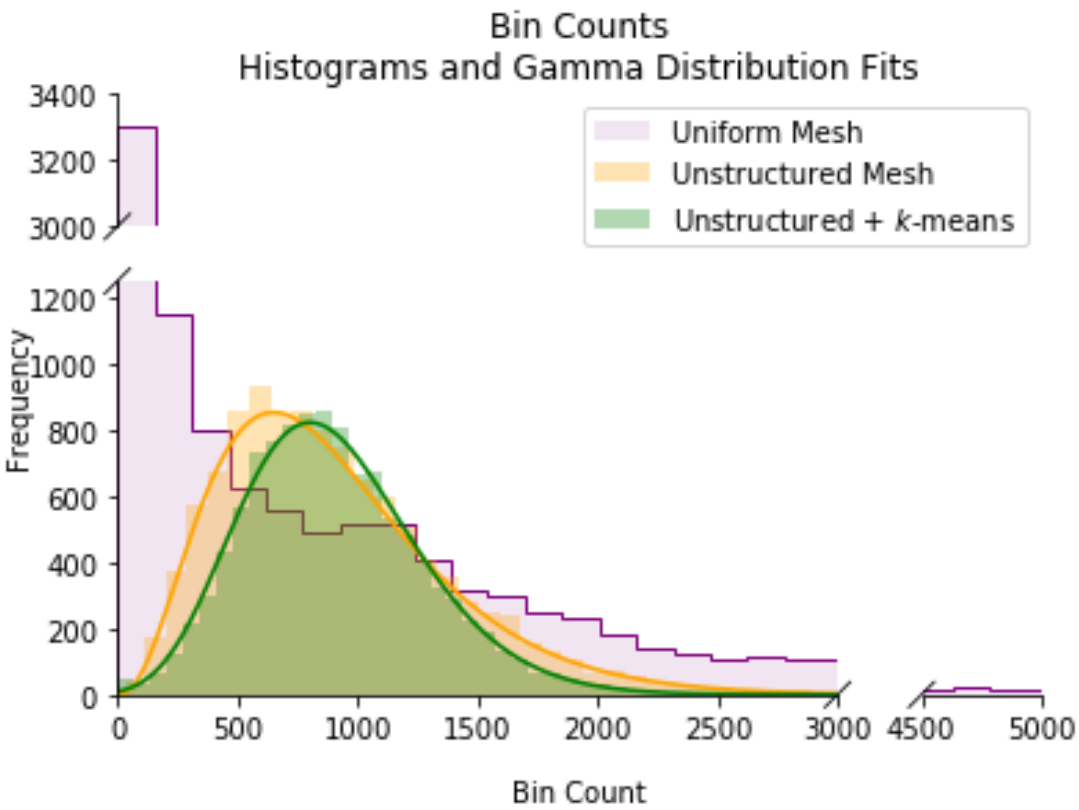
1. If p resides in a simplex of the triangulation:
   - Take the value of p to be the barycentric interpolated value of the simplex vertices.
2. If p does not reside in a simplex:
   - Take the value of p to be the average of its Voronoi cell



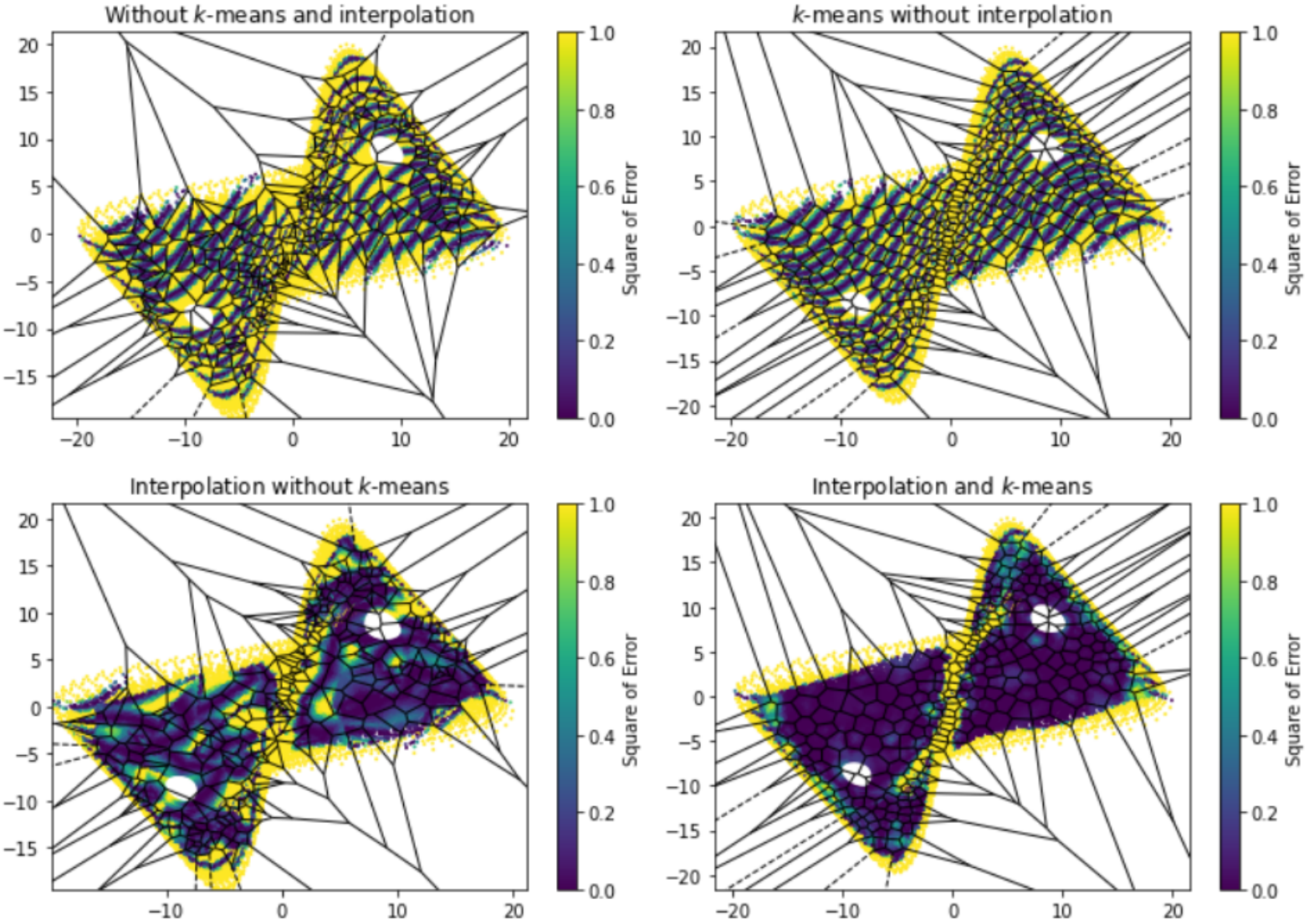


k-means clustering

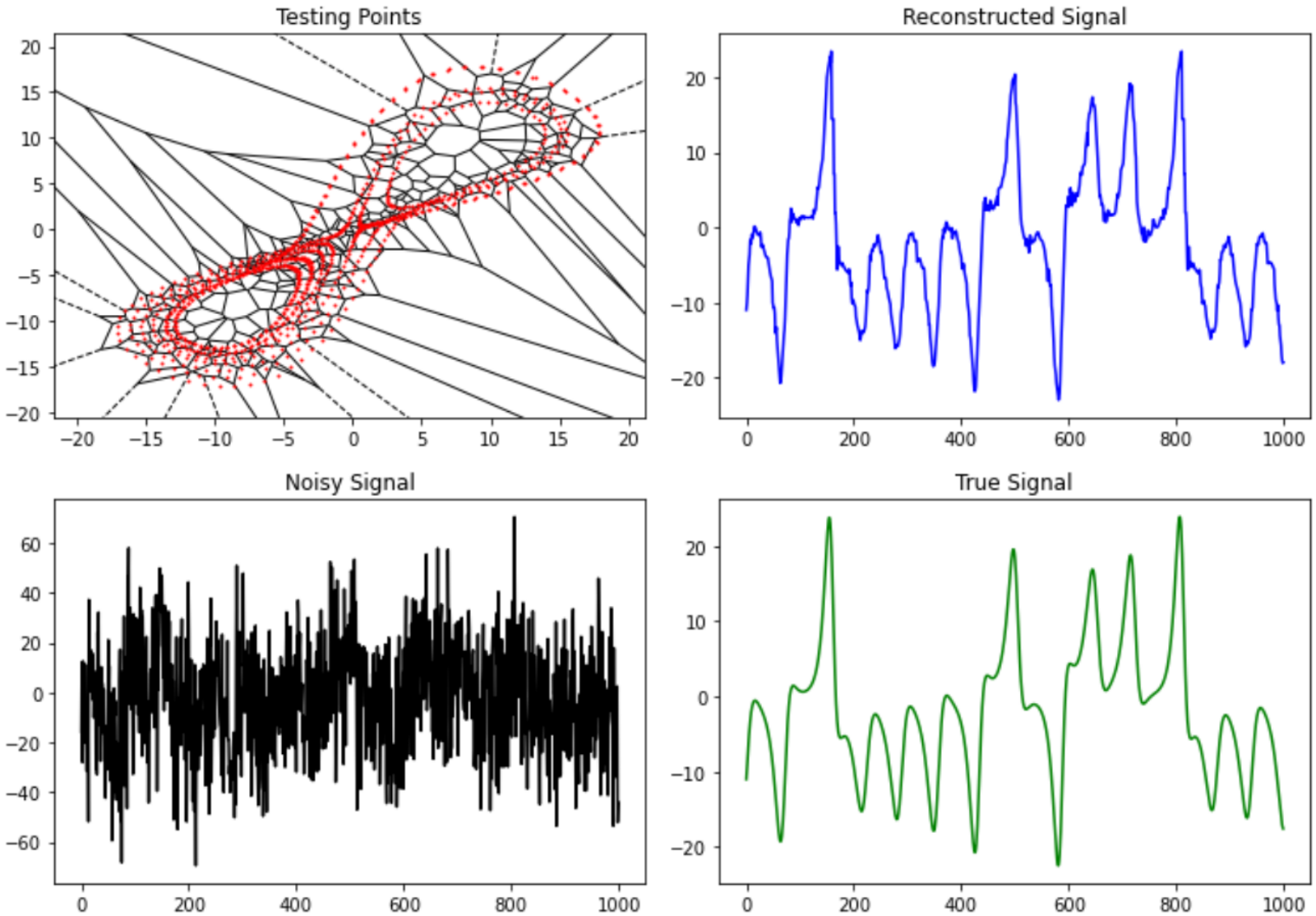
- Instead of choosing the cell representatives randomly, we can use the k-means clustering algorithm.
  - Iteratively selects new cell representatives to be their cell's center of mass.



Effect of interpolation and k-means clustering



Lorenz Signal Reconstruction



Results: Method Evaluation

- Visually confirming success
- Quantitatively confirming success
- Areas of high error

How We Evaluate Our Algorithm

To evaluate our algorithm, we run it on known systems as test data.

1. Start with a clean original signal $Y(t)$
2. Add noise to the original signal to get $Y'(t)$
3. Use our algorithm to get a reconstruction $Y^*(t)$ similar to the the original signal

How We Evaluate Our Algorithm

To evaluate our algorithm, we run it on known systems as test data.

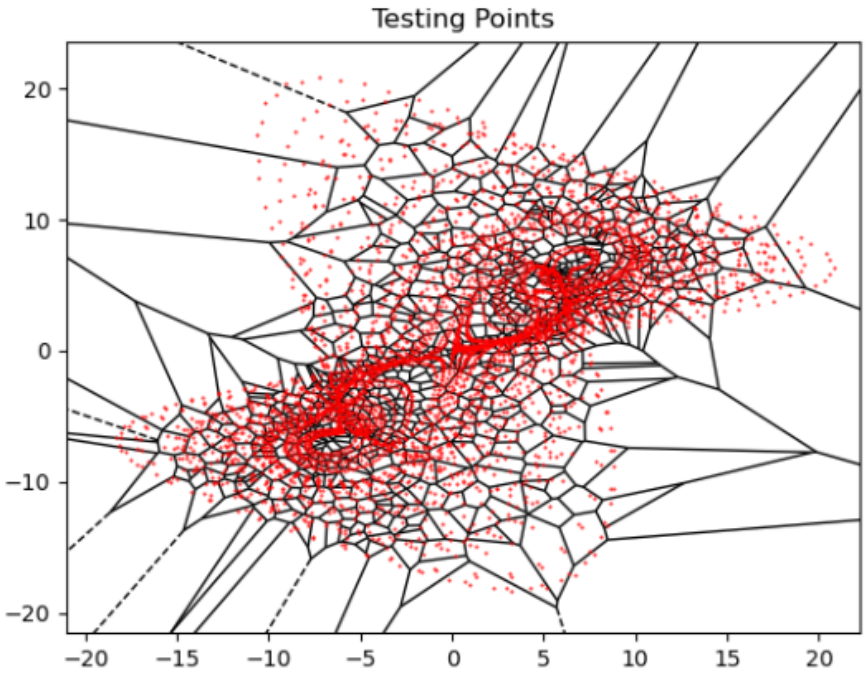
1. Start with a clean original signal $Y(t)$
2. Add noise to the original signal to get $Y'(t)$
3. Use our algorithm to get a reconstruction $Y^*(t)$ similar to the the original signal

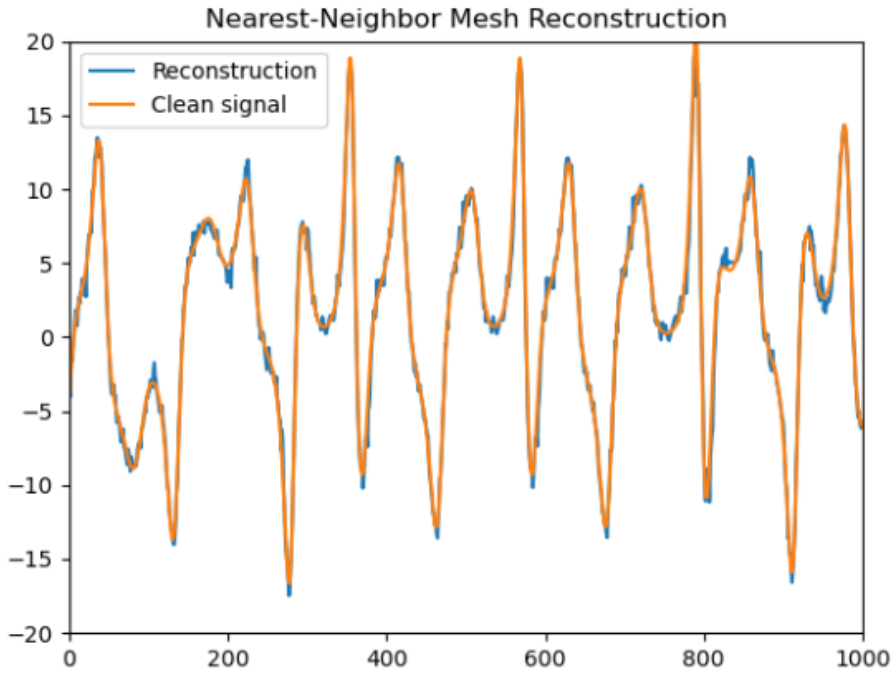
How do we know our reconstruction is accurate?

- **Visually:** $Y^*(t)$ looks like the $Y(t)$.
- **Quantitatively:** as we add data, the error between $Y^*(t)$ and $Y(t)$ linearly converges to zero on a log-log plot.

Visual Evidence of Success: Chen System

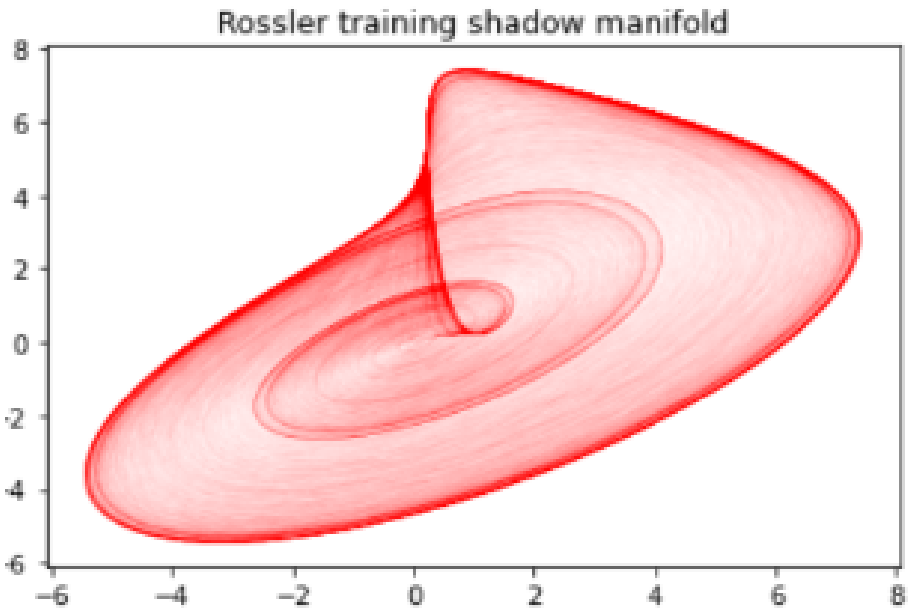
Below is the shadow manifold and signal reconstruction of our algorithm applied to another attractor, the Chen system.

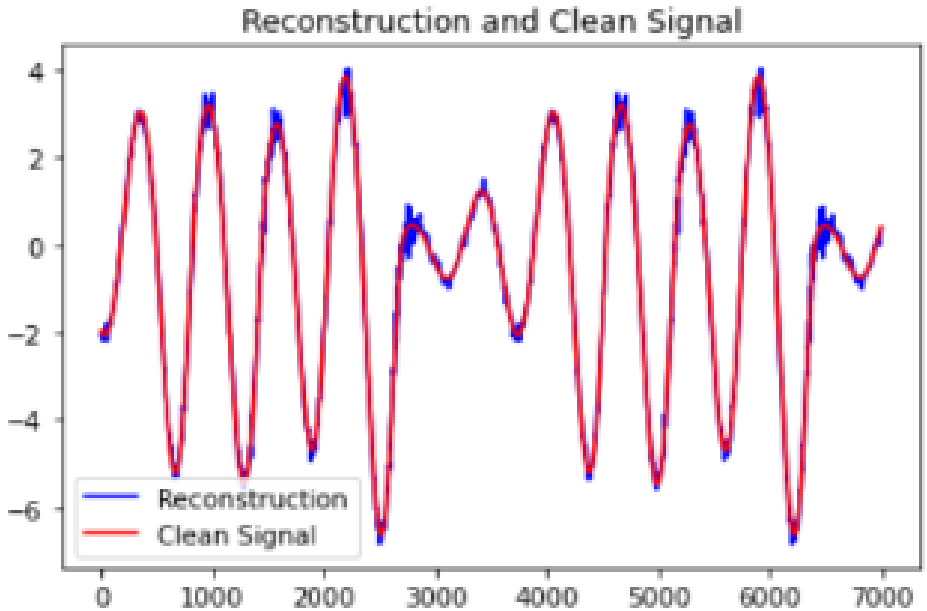




Visual Evidence of Success: Rossler System

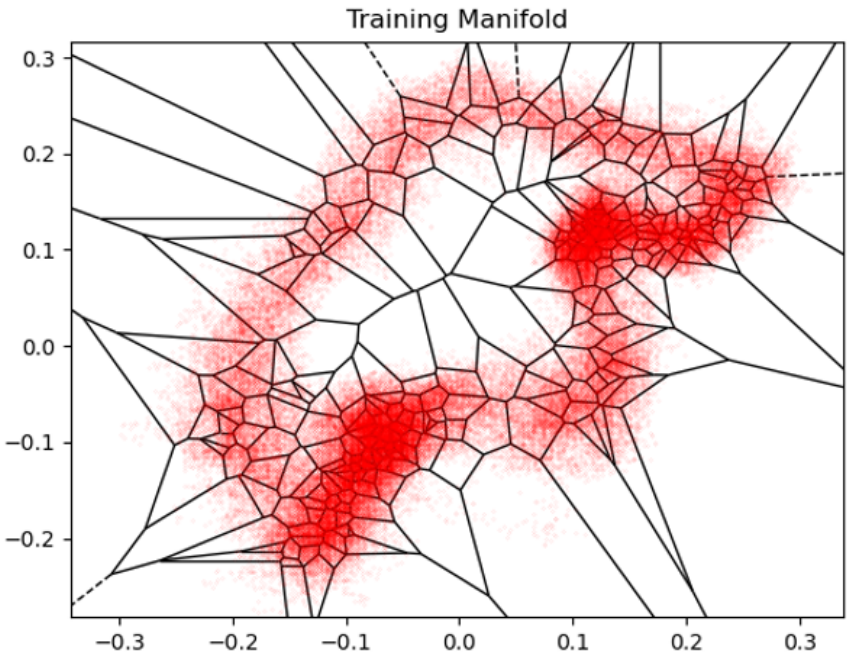
Below is the shadow manifold, noisy signal, and reconstruction of our algorithm applied to the Rossler system.

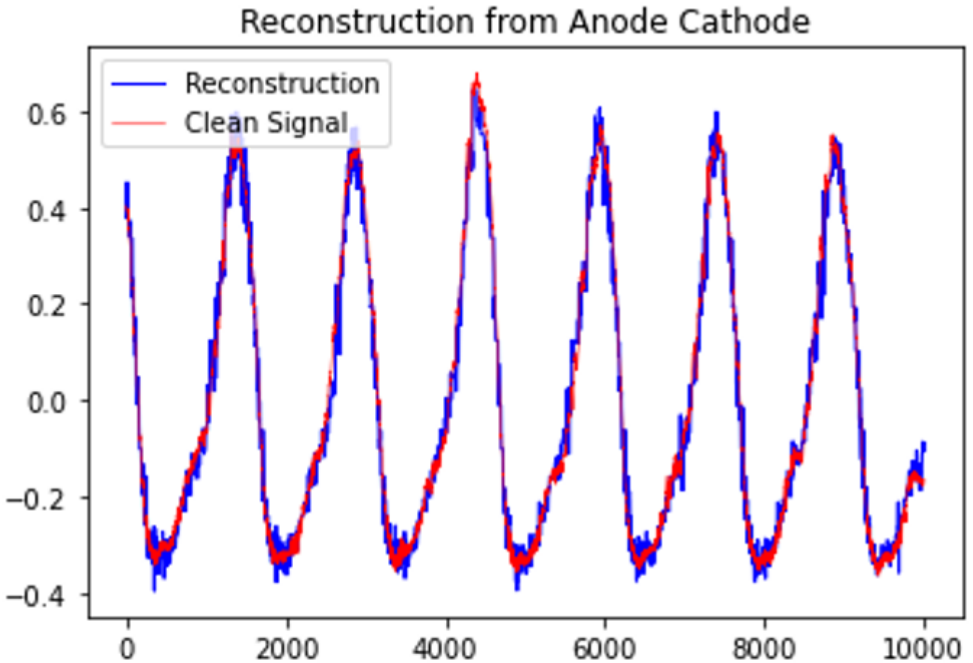




Visual Evidence of Success: HET System!

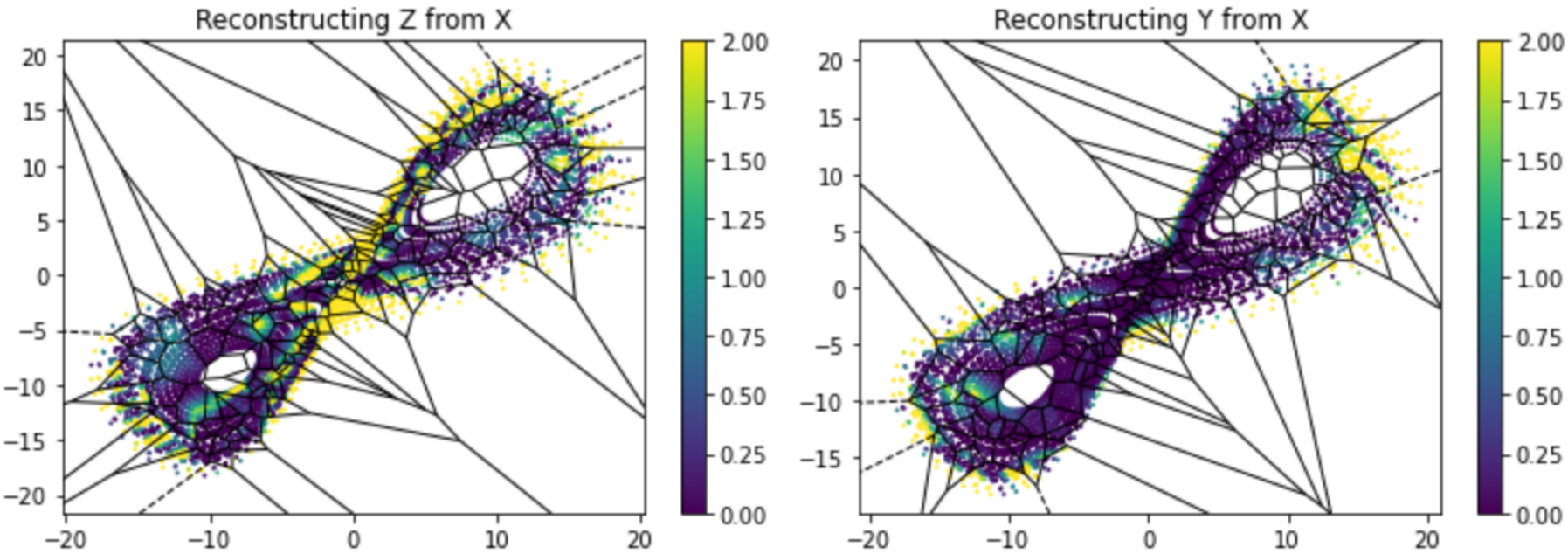
Shadow manifold of Anode+Cathode signal and reconstruction of Cathode Pearson signal





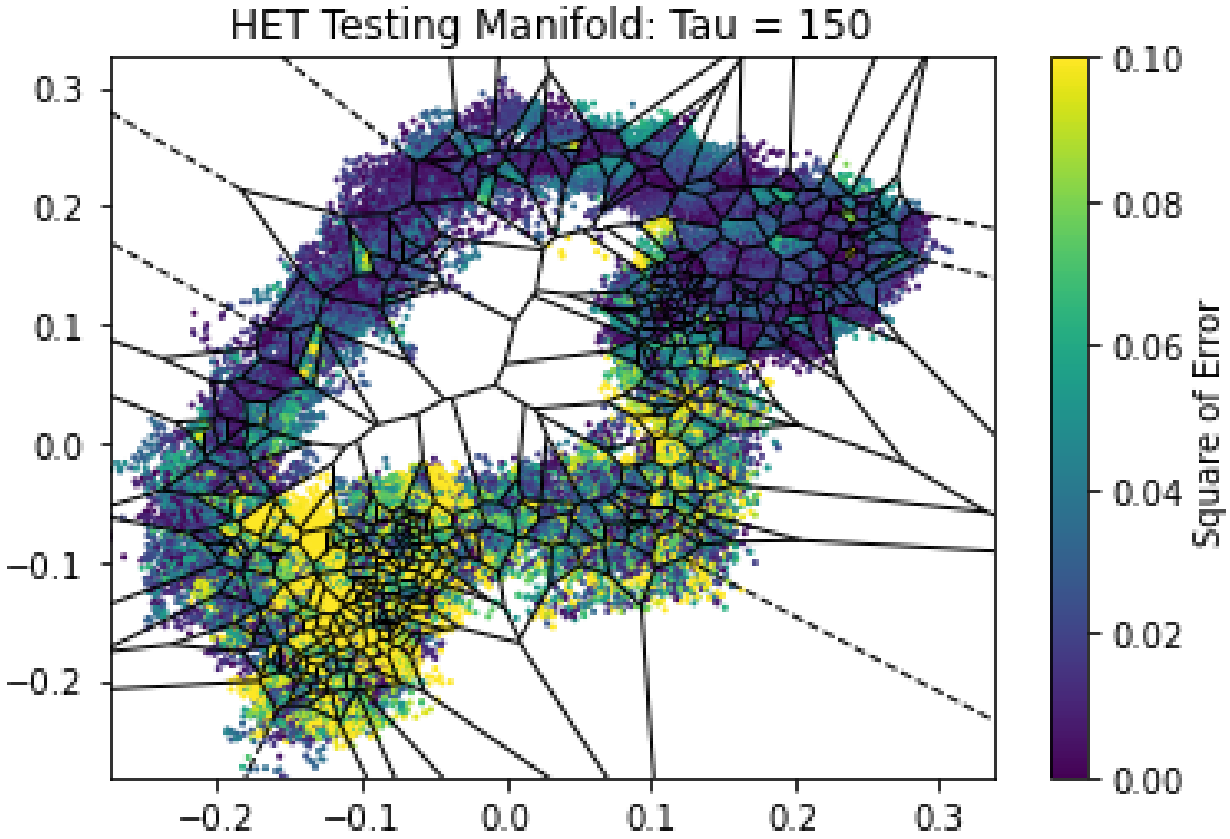
Lorenz System Errors

- Highest error arises at the crossings and outlier points, when reconstructing Z
- Highest error arises mainly at outlier points, when reconstructing Y



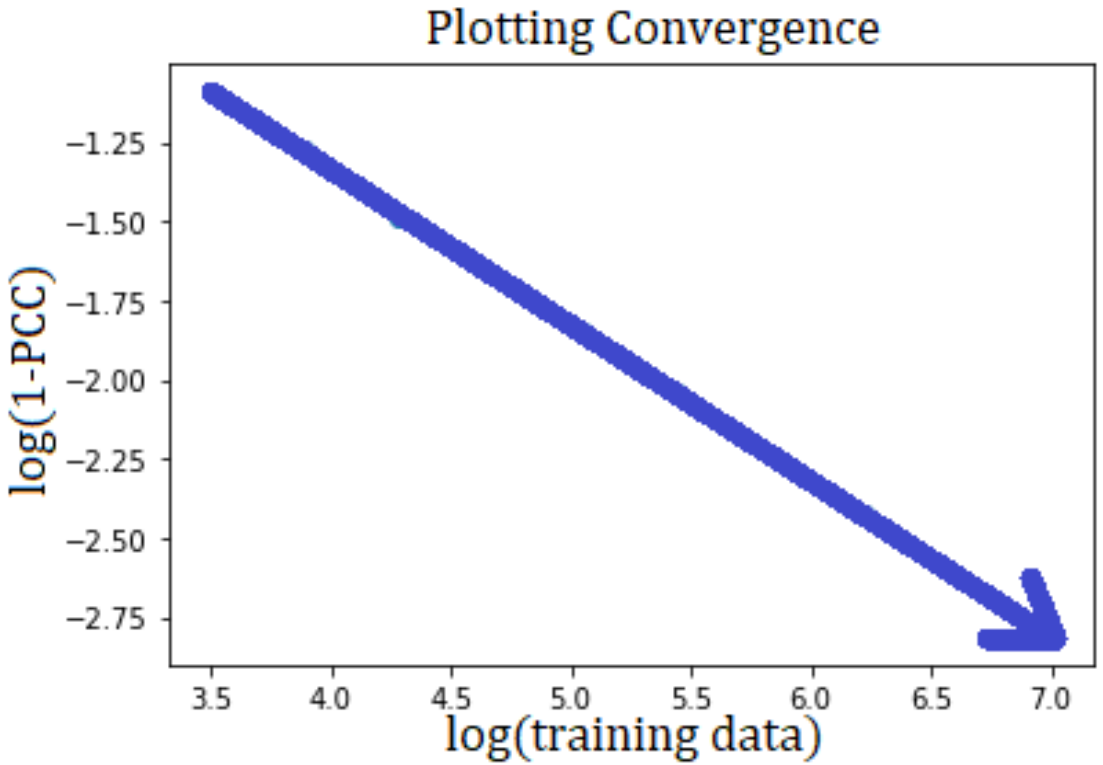
HET System Errors

- Highest error occurs at a crossing in the data.
- Crossings cause points from different parts of the attractor's period to be grouped into the same cell
  - May be improved by increasing τ or d



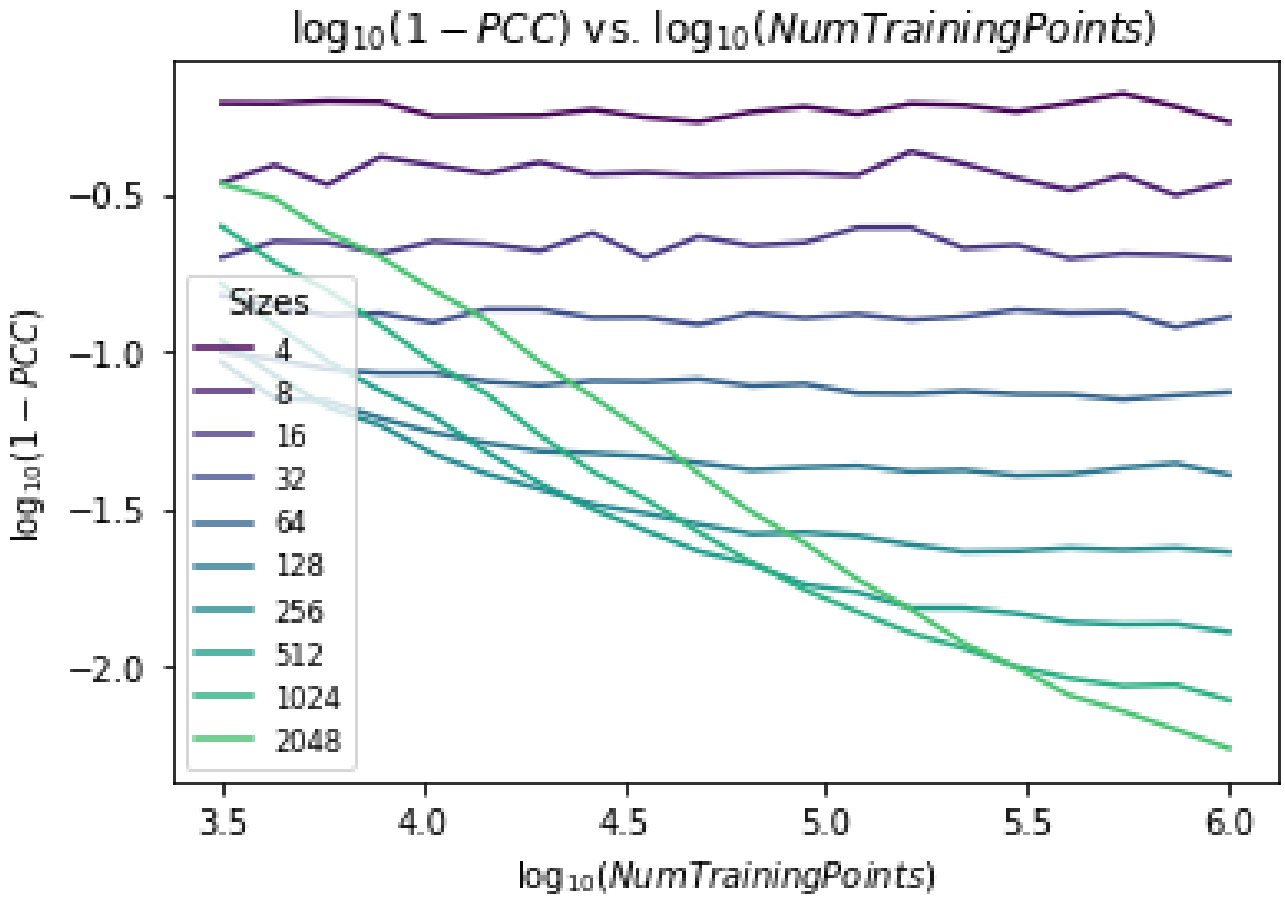
Quantitative Evidence of Success

- We observe this as a linear relationship between training data and error coefficient on a log-log scale.



Convergence Plots on Lorenz Data

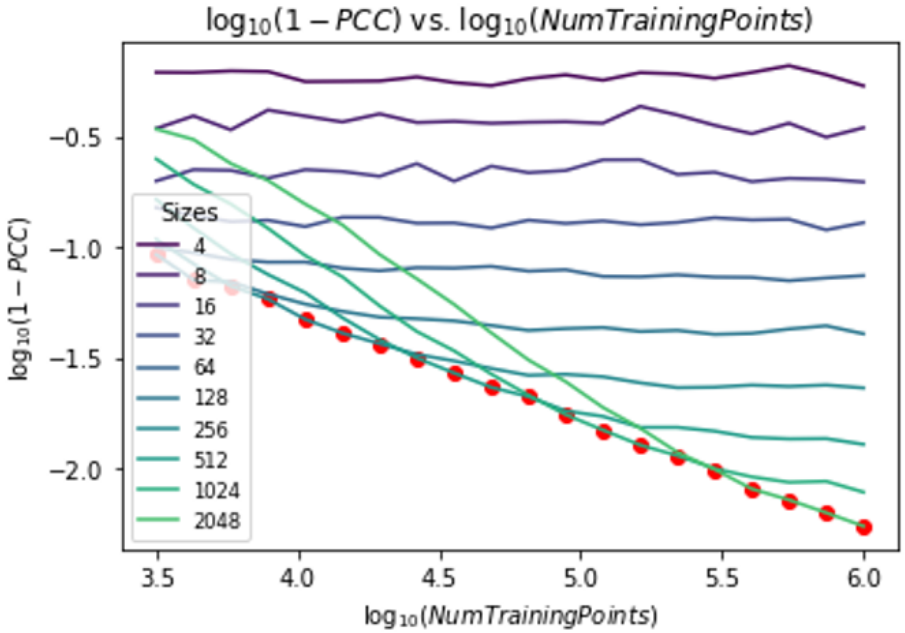
Using the Lorenz system as test data, we fix the number of cells as we reconstruct Lorenz Z.

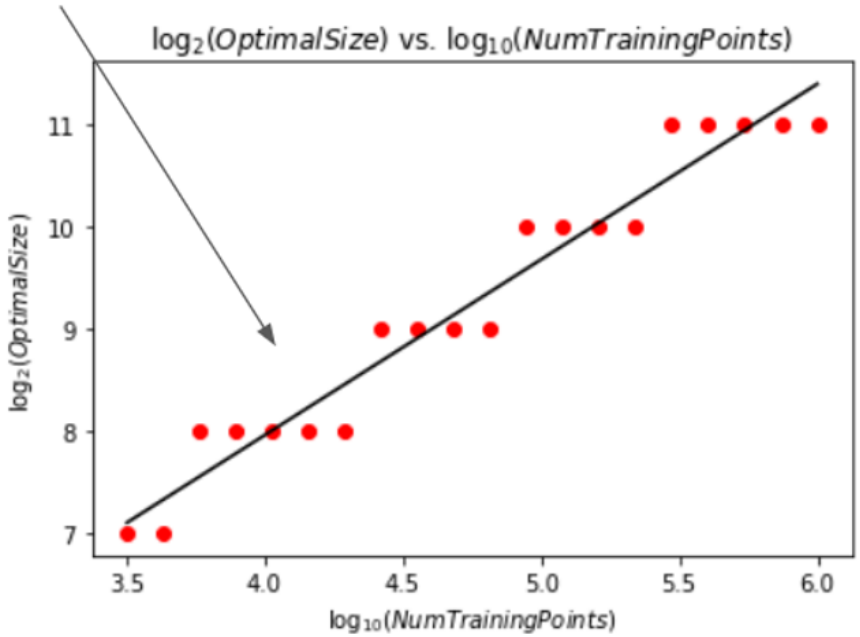


Optimizing Number of Cells on Lorenz

Using information from the fixed # cells plots, we realize a linear relationship between the training data and the number of cells.

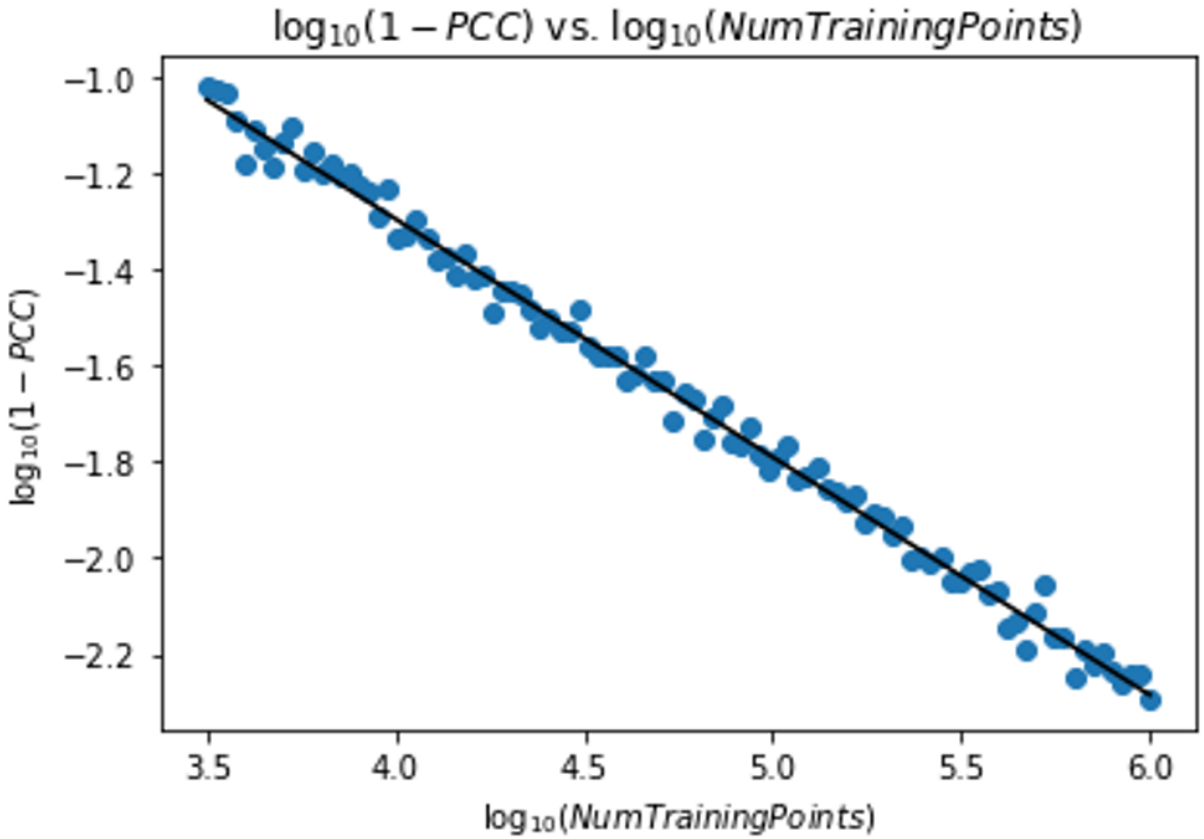
$$\log_2(Size) = 1.72 \cdot \log_{10}(Num) + 1.08$$





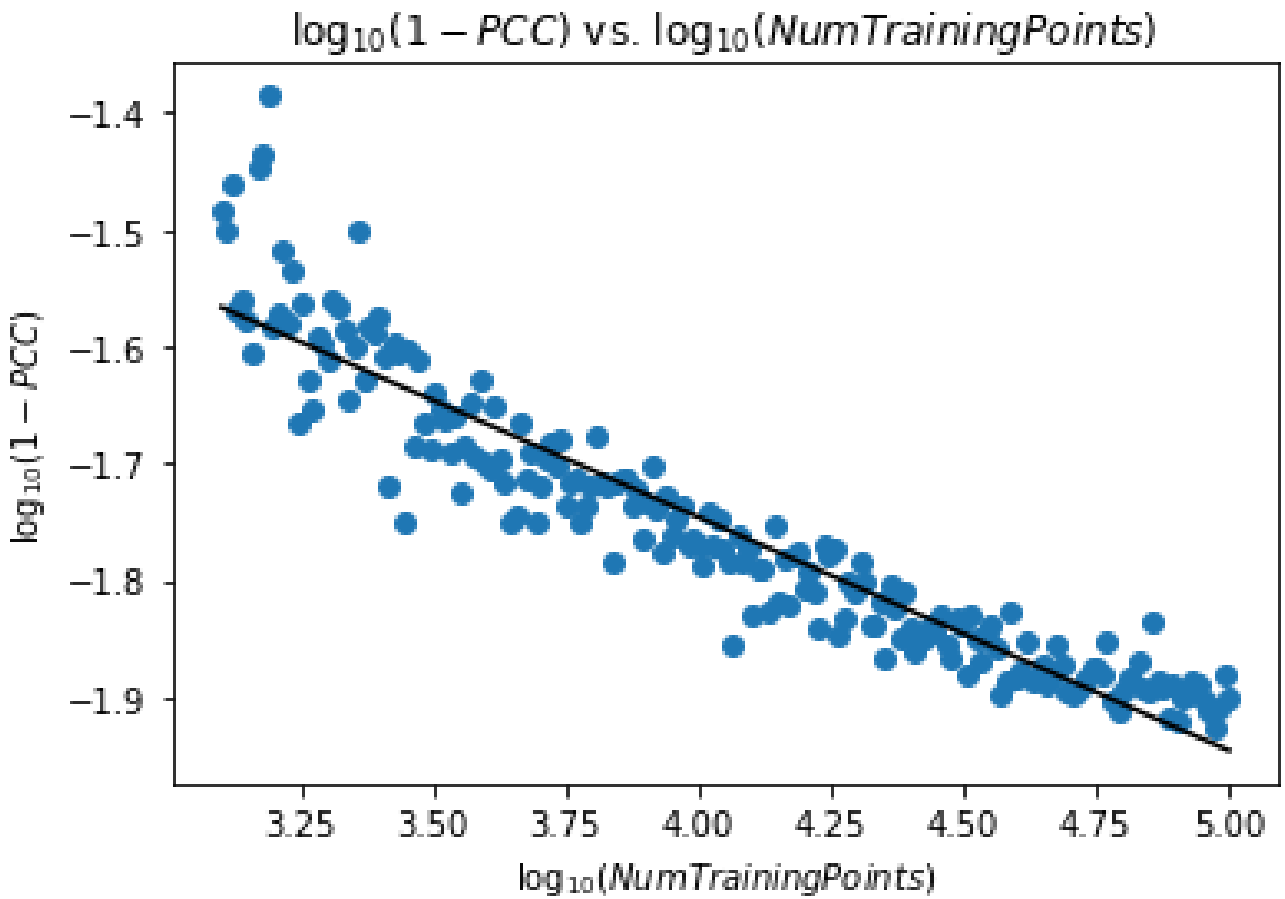
Quantitative Evidence of Success: Lorenz Reconstruction

This log-log plot shows convergence as the number of training points increases, using shadow embedding dimension 3.



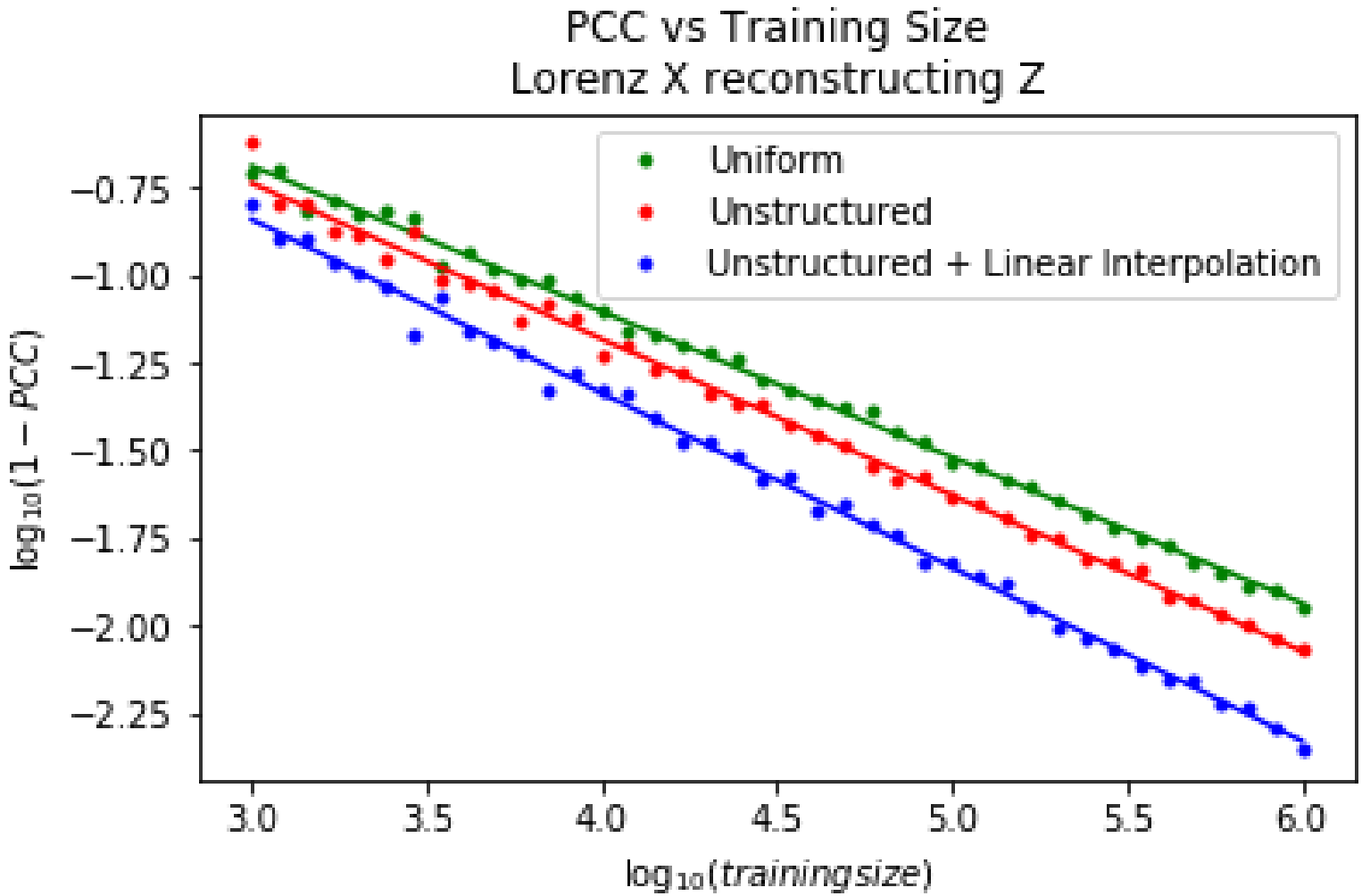
HET Reconstruction

This log log plot is for the reconstruction of Cathode Pearson HET data. Though it doesn't completely show convergence, it does show a promising trend, especially for the limited amount of data.



Comparison to Uniform Mesh

We see slightly faster convergence with less residual error using the unstructured mesh, and even more so including interpolation.



Conclusions

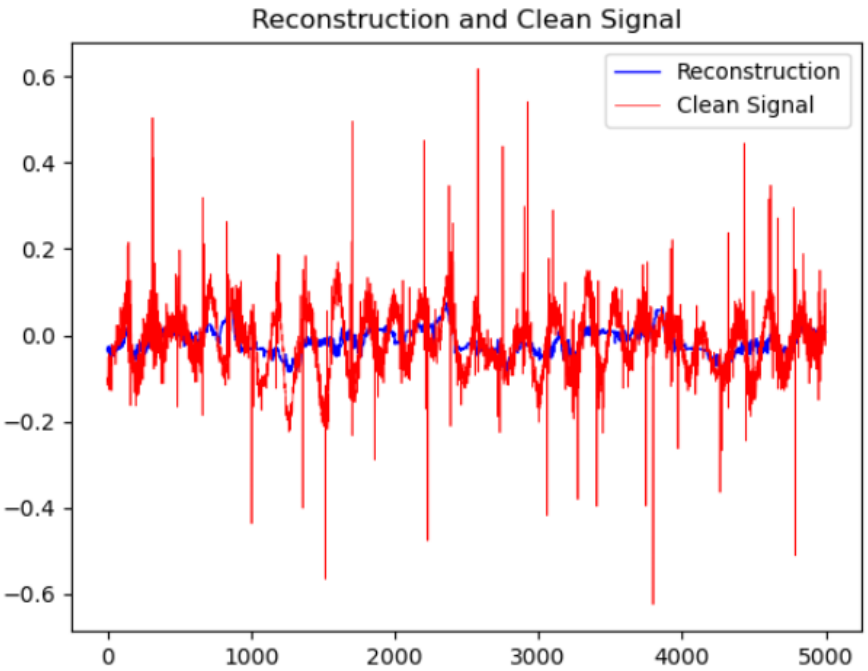
- Deliverables
- Future Work

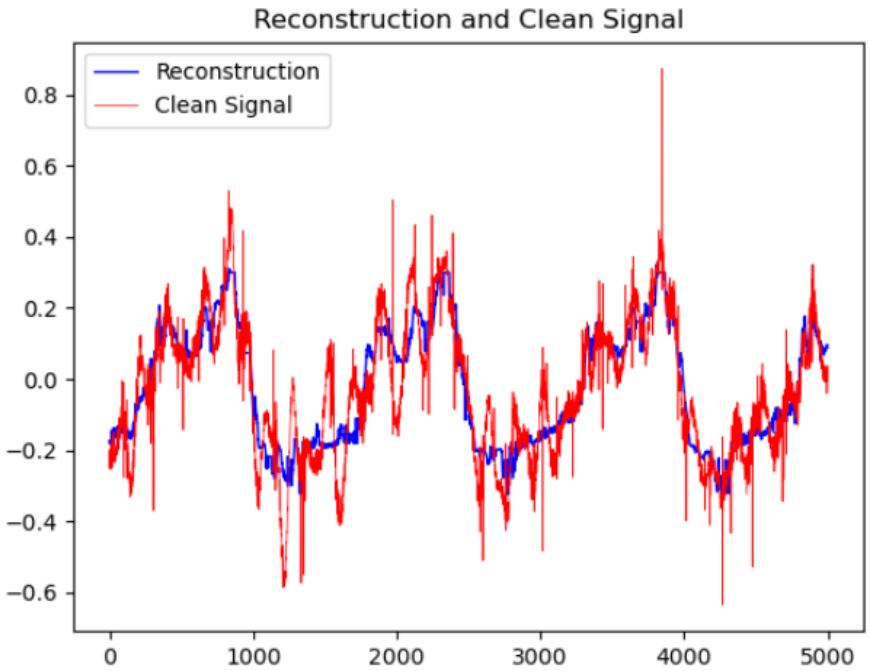
Final Deliverables

- Unstructured mesh denoising algorithm
- Successful HET signal reconstructions
- Thorough analysis of convergence and errors

Recommended Future Work

- Use this method to simultaneously denoise both signals
- Use multiple clean signals to recover more features of one noisy





Questions?

Special thanks to our mentors and sponsors!