

# Discovery of Lagged Dynamic Mode Decomposition Models through Entropic Regression

Christopher W. Curtis<sup>1</sup>, Daniel Jay Alford-Lago<sup>1,2</sup>, and Erik Bollt<sup>3,4</sup>

<sup>1</sup>Department of Mathematics and Statistics, SDSU

<sup>2</sup>Naval Information Warfare Center

<sup>3</sup>Department of Electrical and Computer Engineering, Clarkson University

<sup>4</sup>Clarkson Center for Complex Systems Science, Clarkson University

## Abstract

In this work, we present an information theoretic method which determines optimal multi-step dynamic mode decomposition models via entropic regression, which is a nonlinear information flow detection algorithm. We test our method over several data sets generated by chaotic attractors and show that we are able to build excellent reconstructions using relatively minimal models. We likewise are able to better identify multiscale features via our models which enhances the utility of dynamic mode decomposition.

## 1 Introduction

Equation free model development has enjoyed a multi-year stretch of continued progress in part by way of the evolution of dynamic mode decomposition (DMD) methods. Beginning in the fluid dynamics community (see [1] for a historical review and key sources), DMD has moved out into almost every area of data driven science, and it has merged itself with every major trend in the data sciences as well, in particular machine learning based methods; see [2, 3, 4].

Likewise, in order to better address the modelling of multiscale data, delay structured DMD models reminiscent of either ARIMA or Takens embedding models have also been explored; see [5, 6, 7, 8, 9, 10]. Across these works, it has become clear that the success of DMD when used to generate models which can be time stepped forward from some set of initial data depends upon finding accurate multi-step Koopman like models for a given time series. However, much of the existing literature has in practice focused on data coming from periodic and quasi-periodic attractors, which

as is proved in [7], can be exactly reconstructed using relatively straightforward lagged DMD models. While [5] shows that Hankel-DMD methods should provide exact determination of the affiliated Koopman operator in ergodic systems, this is not necessarily a practical result. This issue is partially addressed in [10], which coupled an adaptive Hankel DMD method to an autencoding neural network strategy to discover meaningful observables from chaotic time series. However, this method did not provide a fully automatic way to determine the updating of the Takens style embeddings needed to make the method successful. Thus, discovering methods which build better lagged DMD models has a number of potential downstream applications.

To improve then on this situation, we treat discovering lagged DMD models as a question of optimal model discovery. While there are a plethora of approaches to this problem, the one that we find most impactful is from [11], which treats the problem of finding the most likely model as the one which is best at describing the flow of information between time scales. Information flow is measured here via a generalization of transfer entropy [12], called *causation entropy* [13], coupled with an algorithmic approach to model discovery called *entropic regression* [11], making our approach one which seeks to find those models which provide the most information about evolving time series. In some sense then, this work is also a natural generalization of the Akaike Information Criteria [14] used to develop parsimonious ARIMA models in statistics.

As we show, our method, which we dub the entropic-regression-dynamic-mode decomposition (ERDMD), is able to discover non-uniform lagged DMD models which provide accurate time-stepping schemes from data coming from a variety of chaotic attractors. This realizes the ambition of the results in [5] while pointing towards future improvements in the algorithm presented in [10]. That said, we also find limitations by way of our results on the Kuramoto–Sivashinsky equation, which shows even optimally lagged DMD models struggle in of themselves to accurately reconstruct higher dimensional strange attractors. Future work in this direction by way of coupling our current method with the work of [10] should address this shortcoming though. Likewise, as noted in [11], a major motivation for using information theory in model discovery is that it supports robust model discovery from noisy data, which is also a natural topic of future inquiry.

The structure of the paper is as follows. In Section 2, we provide a brief introduction to higher order DMD and entropic regression. We then present our central algorithm, the entropic regression DMD method. In Section 3, we present the results of our method across three different chaotic dynamical systems. Finally in Section 4, we provide a discussion of the method and further elaborate on future directions which would follow from the present work.

## 2 Entropic Regression DMD

The results in this work are a merging of the higher-order DMD (HODMD) method of [6], and the entropic regression technique for network detection and model construction found in [13, 11]. We now briefly explain both results, and then we show how to bring them together in order to build accurate, yet minimal time lag models.

### 2.1 Higher-Order DMD

Originally presented in [6], though see also the HAVOK method in [15] and Hankel DMD method of [5], we generalize the HODMD so as to keep each lagged model separate, causal, and formulated over an arbitrary sequence of not necessarily unit increment lags. We suppose that we have the maximum lag of  $d$  time steps. We likewise suppose that we have the choice of say  $N_L$  lags as  $l_c = \{l_1, l_2, \dots, l_k, \dots, l_{N_L}\}$ , with  $1 = l_1 < l_j < l_{j+1} < l_{N_L} < d$ . Given times series  $\{\mathbf{y}_j\}_{j=0}^{N_T}$ , we then define the matrices

$$\mathbf{Y}_{+,d} = (\mathbf{y}_d \cdots \mathbf{y}_{N_T}), \quad \mathbf{Y}_- = (\mathbf{y}_0 \cdots \mathbf{y}_{N_T-1}), \quad \mathbf{y}_j \in \mathbb{R}^s,$$

and the  $s \times (N_T - d + 1)$  shifted mask matrices  $\mathbf{M}_l$  where

$$(\mathbf{M}_l)_{mn} = \begin{cases} 0 & m-1 < d-l, \quad m-1 > N_T-l \\ 1 & n=m, \quad d-l \leq m-1 \leq N_T-l \\ 0 & n \neq m, \quad d-l \leq m-1 \leq N_T-l \end{cases}$$

An arbitrary lagged DMD model can then be written as

$$\mathbf{Y}_{+,d} = \sum_{k=1}^{N_L} \mathbf{K}_{l_k} \mathbf{Y}_- \mathbf{M}_{l_k}, \quad (1)$$

where we note that each matrix  $\mathbf{K}_{l_k}$  is  $s \times s$ . Using our standard optimization arguments, we can find each matrix  $\mathbf{K}_{l_j}$  via the critical-point equation

$$\sum_{k=1}^{N_L} \mathbf{K}_{l_k} \mathbf{Y}_- \mathbf{M}_{l_k} \mathbf{M}_{l_j}^T \mathbf{Y}_-^T = \mathbf{Y}_{+,d} \mathbf{M}_{l_j}^T \mathbf{Y}_-^T.$$

Combining these terms across lags leads to the system

$$\mathbf{K}(l_c) \mathbf{Y}_-(l_c) \mathbf{Y}_-(l_c)^T = \mathbf{Y}_{+,d} \mathbf{Y}_-(l_c)^T.$$

where

$$\mathbf{K}(l_c) = \left( \mathbf{K}_{l_{N_L}} \mathbf{K}_{l_{N_L-1}} \cdots \mathbf{K}_1 \right),$$

and

$$\mathbf{Y}_-(l_c) = \begin{pmatrix} \mathbf{Y}_- \mathbf{M}_{l_{N_L}} \\ \mathbf{Y}_- \mathbf{M}_{l_{N_L}-1} \\ \vdots \\ \mathbf{Y}_- \mathbf{M}_1 \end{pmatrix}.$$

Once the model in Equation (1) is built, we run it by coupling outputs to inputs so that for  $j \geq d$ , we have

$$\mathbf{y}_{j+1} = \sum_{k=1}^{N_L} \mathbf{K}_{l_k} \mathbf{y}_{j+1-l_k} \mathbf{M}_{l_k}. \quad (2)$$

While the method sees data within the original given time series, we say the method is performing *reconstruction*. When it begins to iterate over data it has not seen, then we say the model is performing *forecasting*.

In [6], models with distinct matrices  $\mathbf{K}_{l_j}$  for each lag were eschewed for something much closer to what is now called Hankel DMD. While effective, and in several respects simpler, such an approach does not as readily allow for more thoughtful model selection as we explain in the next section. That said, a price is paid for working with separate lag matrices when one wants to look at the corresponding one-step Koopman operator, say  $\mathcal{K}^\delta$  which would be approximated by

$$\mathcal{K}^\delta \approx \mathbf{K}_a \equiv \begin{pmatrix} 0 & I & 0 & \cdots & 0 & 0 \\ 0 & 0 & I & \ddots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & I & 0 \\ \mathbf{K}_{l_{N_L}} & \tilde{\mathbf{K}}_{l_{N_L}-1} & \cdots & \cdots & \tilde{\mathbf{K}}_2 & \mathbf{K}_1 \end{pmatrix} \quad (3)$$

where each  $s \times s$  matrix  $\tilde{\mathbf{K}}_j$  is given by

$$\tilde{\mathbf{K}}_j = \begin{cases} \mathbf{K}_{l_k}, & j = l_k \in l_c \\ \mathbf{0}, & j \neq l_k \in l_c \end{cases}$$

Finding the eigenvalues of  $\mathbf{K}_a$  is equivalent to finding the roots of the polynomial  $p_a(z)$  where

$$p_a(z) = \det \left( \sum_{k=0}^{N_L-1} \mathbf{K}_{l_{N_L-k}} z^{l_{N_L}-l_{N_L-k}} - z^{l_{N_L}} \mathbf{I} \right). \quad (4)$$

While not useful in a numerical context, this formula will prove useful when characterizing the eigenvalues of  $\mathbf{K}_a$  relative to the choice of lags  $l_c$ .

## 2.2 Determining Information Flow and Model Discovery through Entropic Regression

Given two time series, say  $\{X_j\}_{j=1}^{N_T}$  and  $\{Y_j\}_{j=1}^{N_T}$ , it is a foundational question to determine if one time series *causes* the other. Said another way, can we find quantitative methods which determine how one time series might drive or ultimately explain the behavior of another?

Motivated by the now celebrated *Granger causality* test, *cf.* [16], in linear time series, [12] introduced the notion of *transfer entropy* to determine the causal relationship between two nonlinear time series. The transfer entropy from  $X_j$  to  $Y_j$ , say  $T_{X \rightarrow Y}(j)$  is defined in [12] to be

$$T_{X \rightarrow Y}(j) = H(Y_{j+1}|Y_j) - H(Y_{j+1}|Y_j, X_j), \quad (5)$$

$$= I(Y_{j+1}, X_j|Y_j). \quad (6)$$

where  $H(\cdot)$  measures the entropy of a random variable and  $I(\cdot, \cdot)$  is the mutual information between two random variables defined via the equation

$$I(X, Y) = H(X) - H(X|Y). \quad (7)$$

Thus the transfer entropy  $I(Y_{j+1}, X_j|Y_j)$  measures the conditional mutual information with conditioning over  $Y_j$ . Note, if  $Y_{j+1}$  is independent of  $X_j$ , then  $H(Y_{j+1}|Y_j, X_j) = H(Y_{j+1}|Y_j)$  so that  $T_{X \rightarrow Y}(j) = 0$ . This initial concept of transfer entropy has given rise to a host of modifications and improvements, the most relevant for our purposes being that of *causation entropy* [13] and *entropic regression* (ER) [11]. Causation entropy provides a natural extension of transfer entropy to networks in which a central question is the determination of information flow between network nodes. Central to its computation over a network is the separation of the algorithm, after some INITIALIZE phase, into a BUILD and PRUNE phase. The PRUNE phase in particular removes nodes which would otherwise provide false-positive connections for information flow in the network, thereby providing a critical mechanism for developing accurate pictures of information flow among several processes.

Building on this three stage methodology, ER develops models of time series from dictionaries of basis functions, say  $\{\phi_l\}_l^{N_m}$ . This happens as well in a INITIALIZE, BUILD, and then PRUNE phase. After choosing some reasonable initial basis function and corresponding model, The BUILD phase is done by fixing a state, say  $\mathbf{Y}_s$ , which is modelled by the current model, say  $\mathbf{M}_c$ , and a test model, say  $\mathbf{M}_t^{(j)}$ , where  $\mathbf{M}_t^{(j)}$  differs from  $\mathbf{M}_c$  through the inclusion of a basis function, say  $\phi_j$ , not already used in constructing  $\mathbf{M}_c$ . We then find a potential model update by computing

$$\tilde{I} = \operatorname{argmax}_j I(\mathbf{Y}_s, \mathbf{M}_t^{(j)} | \mathbf{M}_c),$$

so that, if  $\tilde{I} > 0$  in a stastically significant sense, we choose those model updates which most increase the degree of information that one model provides over another regards to a state  $\mathbf{Y}_s$ . Once we have exhausted the family of basis functions, ER commences the PRUNE phase, in which for the current model  $\mathbf{M}_c$  we generate  $\mathbf{M}_t^{(j)}$  by removing basis functions, say  $\phi_j$ , from  $\mathbf{M}_c$ , and then compute

$$\tilde{i} = \operatorname{argmin}_j I \left( \mathbf{Y}_s, \mathbf{M}_t^{(j)} \middle| \mathbf{M}_c \right),$$

so that if  $\tilde{i} = 0$  in a statistically significant way, then we reduce  $\mathbf{M}_c$  to the chosen  $\mathbf{M}_t^{(j)}$ .

### 2.3 Entropic-Regression-Dynamic-Mode Decomposition

To augment the HODMD method, we now merge it with ER. To do this, fixing some maximum choice of lag  $d$ , we take as our desired state  $\mathbf{Y}_s \equiv \mathbf{Y}_d^+$ . Assuming some already chosen set of lags, say  $l_c$ , we can add to  $l_c$  some not already chosen lag, say  $l_j$ , to generate the set of test lags  $l_t$ . We can then use HODMD to generate a given model  $\mathbf{M}_c \equiv \mathbf{K}(l_c)\mathbf{Y}_-(l_t)$  and a proposed model  $\mathbf{M}_t^{(j)} \equiv \mathbf{K}(l_t)\mathbf{Y}_-(l_t)$ . Following then the ER framework, there are three stages to our ERDMD method, which are

1. **INITIALIZE:** We initialize our choice of lags  $l_c = \{1\}$  and corresponding HODMD matrix  $\mathbf{K}_1$  which is found via the typical DMD method.
2. **BUILD:** Given some choice of lags  $l_c$ , we then look at the proposed list  $l_t = l_c \cup \{l_j\}$  and then find those lagged models, say  $\mathbf{M}_t^{(j)} \equiv \mathbf{K}(l_t)\mathbf{Y}_-(l_t)$  which provide the most information relative to the prior choice of lagged model, say  $\mathbf{M}_c \equiv \mathbf{K}(l_c)\mathbf{Y}_-(l_t)$ , i.e. we find

$$\tilde{I} = \operatorname{argmax}_j I \left( \mathbf{Y}_d^+, \mathbf{K}(l_t)\mathbf{Y}_-(l_t) \middle| \mathbf{K}(l_c)\mathbf{Y}_-(l_t) \right). \quad (8)$$

If this value is larger than zero in a statistically significant way (measured through shuffle testing), then the model is updated to the corresponding model  $\mathbf{M}_t^{(j)}$ ; see Figure 1 for an illustration of this process.

3. **PRUNE:** We finally prune by testing whether each chosen lag is necessary relative to the other choices we have made during the build phase. Thus, for a given model  $\mathbf{M}_c \equiv \mathbf{K}(l_c)\mathbf{Y}_-(l_t)$ , we generate  $\mathbf{M}_t^{(j)} \equiv \mathbf{K}(l_c \setminus l_j)\mathbf{Y}_-(l_t)$  and then find

$$\tilde{i} = \operatorname{argmin}_j I \left( \mathbf{Y}_d^+, \mathbf{M}_t^{(j)} \middle| \mathbf{M}_c \right). \quad (9)$$

If this value is close to zero in a statistically significant way (measured through shuffle testing), then model is updated to the corresponding  $\mathbf{M}_t^{(j)}$ .

$$\begin{aligned}
\mathbf{M}_c : \quad & \begin{array}{c} \text{pink box} \\ \mathbf{y}^{(+)} \end{array} = \begin{array}{c} \text{yellow box} \\ \mathbf{K}_1 \end{array} \begin{array}{c} \text{pink box} \\ \mathbf{y}_1^{(-)} \end{array} + \cdots + \begin{array}{c} \text{yellow box} \\ \mathbf{K}_{l_k} \end{array} \begin{array}{c} \text{pink box} \\ \mathbf{y}_{l_k}^{(-)} \end{array} \\
\\
\mathbf{M}_t^{(j)} : \quad & \begin{array}{c} \text{pink box} \\ \mathbf{y}^{(+)} \end{array} = \begin{array}{c} \text{yellow box} \\ \tilde{\mathbf{K}}_1 \end{array} \begin{array}{c} \text{pink box} \\ \mathbf{y}_1^{(-)} \end{array} + \cdots + \begin{array}{c} \text{yellow box} \\ \tilde{\mathbf{K}}_{l_k} \end{array} \begin{array}{c} \text{pink box} \\ \mathbf{y}_{l_k}^{(-)} \end{array} \\
& + \begin{array}{c} \text{yellow box} \\ \tilde{\mathbf{K}}_{l_k+j} \end{array} \begin{array}{c} \text{pink box} \\ \mathbf{y}_{l_k+j}^{(-)} \end{array}
\end{aligned}$$

Figure 1: The BUILD stage of the ERDMD algorithm. Here, we start from the given model  $\mathbf{M}_c$  which represents the choice of lags  $(1, \dots, l_k)$  with correspondig matrices  $(\mathbf{K}_1, \dots, \mathbf{K}_{l_k})$ , and we then find a proposed model  $\mathbf{M}_t^{(j)}$  which maximizes  $I(\mathbf{y}^+, \mathbf{M}_t^{(j)} | \mathbf{M}_c)$ , or the information gain in using the proposed model relative to the given model to anticipate the next time steps represented by  $\mathbf{y}^{(+)}$ .

Note, we always have  $1 \in l_c$  since this makes all subsequent lags improvements on the basic DMD approach. This process is formalized and detailed in Algorithm 1.

### 3 Results

To study the efficacy of our method, we examine its use over numerically generated data from various chaotic dynamical systems. In each case, for a given time series  $\{\mathbf{y}_j\}_{j=0}^{N_T}$  we choose a maximum lag  $d$  and then use a derived model to reconstruct the original time series after the  $d^{th}$  step, i.e.  $\{\mathbf{y}_j\}_{j=d}^{N_T}$ . In each case, we compare the model generated by our proposed ERDMD method to the full HODMD method. While this is of course not exhaustive with regards to model performance comparison, we refer the reader to [10], which looks at the performance of the Hankel DMD method on a similar class of problems where we see it generally fail outright without further modification. Thus, within its appropriate method class, we have a clear sense that the ERDMD method is performing well across its most nearby competitors.

#### 3.1 Lorenz-63

We now examine how well our method performs on the Lorenz-63 system, given by the equations

$$\begin{aligned}\dot{y}_1 &= \sigma(y_2 - y_1) \\ \dot{y}_2 &= y_1(\rho - y_3) - y_2 \\ \dot{y}_3 &= y_1y_2 - \beta y_3\end{aligned}$$

with  $\sigma = 10$ ,  $\rho = 28$ , and  $\beta = 8/3$ . These parameter choices ensure that trajectories are pulled onto a strange attractor and exhibit chaotic dynamics. We test our method on numerically generated data. Time stepping is done with standard Runge–Kutta 4. For our numerics, the time step is  $dt = .01$ , and we run the simulation from  $0 \leq t \leq 22$ .

We look at data for  $20 \leq t \leq 22$ , with  $d = 150$ , corresponding to 1.5 units of non-dimensional time. We then compare our ER-DMD model to a model using all possible lags for  $21.5 \leq t \leq 22$ . The ERDMD algorithm converges to  $l_c = \{1, 149\}$ . The results of running the two models in both reconstruction and forecasting modes are seen in Figure 2. In the error plots in Figure 2 (b), we see that throughout the reconstruction region, the HODMD is categorically more accurate than the ERDMD method by several orders of magnitude. However, practically speaking, the approximation provided by the ERDMD is still quite good. Further, looking beyond the reconstruction



---

**Algorithm 1** ERDMD Method

---

```

1: procedure INITIALIZE
2:   Set  $l_c = \{1\}$  and  $l_r = \{2, \dots, d\}$ .
3:   Find  $\mathbf{K}_1 = \arg \min_{\mathbf{K}} \|\mathbf{Y}_1^+ - \mathbf{K}\mathbf{Y}_-(l_c)\|_F$ . Set  $\mathbf{K}(l_c) = (\mathbf{K}_1)$ .
4: end procedure
5: procedure BUILD
6:   while  $l_r \neq \{\emptyset\}$  do
7:     Given  $l_c = \{1, l_1, \dots, l_j\}$ ,  $\mathbf{K}(l_c)$ , and  $l_r = \{2, \dots, d\} \setminus l_c$ 
8:     for  $l_{j+1} \in l_r$  do
9:       Define  $l_t = l_c \cup \{l_{j+1}\}$  and find

$$\mathbf{K}(l_t) = \arg \min_{\tilde{\mathbf{K}}_1, \tilde{\mathbf{K}}_{l_1}, \dots, \tilde{\mathbf{K}}_{l_{j+1}}} \left\| \mathbf{Y}_{l_{j+1}}^+ - \tilde{\mathbf{K}}(l_t) \mathbf{Y}_-(l_t) \right\|_F.$$

10:    end for
11:    Choose  $l_{j+1}$  and the corresponding  $l_t$  and  $\mathbf{K}(l_t)$  to maximize

$$I(\mathbf{Y}_d^+, \mathbf{K}(l_t) \mathbf{Y}_-(l_t) | \mathbf{K}(l_c) \mathbf{Y}_-(l_t))$$

12:    If choice is statistically significant (using shuffle test), update  $l_c$ 
    and  $\mathbf{K}(l_c)$ .
13:  end while
14: end procedure
15: procedure PRUNE
16:   Given  $l_c = \{1, l_1, \dots, l_{N_L}\}$ , set  $S \equiv \text{True}$ 
17:   while  $S$  do
18:     for  $l_j \in l_c$  do
19:       Define  $l_t = \{1, l_1, \dots, l_{N_L}\} \setminus \{l_j\}$ 
20:       Compute

$$I(\mathbf{Y}_{l_{N_L}}^+, \mathbf{K}(l_t) \mathbf{Y}_-(l_t) | \mathbf{K}(l_c) \mathbf{Y}_-(l_t)).$$

21:     end for
22:     Choose  $l_t$  corresponding to minimum information.
23:     if minimum is statistically insignificant then
24:       Prune corresponding  $l_j$  from  $l_c$  and  $\mathbf{K}(l_c)$ .
25:     else
26:        $S \equiv \text{False}$ 
27:     end if
28:   end while
29: end procedure

```

---

to the forecast regime, we see the ERDMD model does a generally better job beyond the data, thereby implying that the HODMD model is probably overfitting.

To this end, we can also compare the models generated by ERDMD and HODMD. Looking at the norms of the various lag matrices in Figure 3 we see that the ERDMD model produces a much sparser model with significantly larger overall matrix norms. Likewise, the relative continuity of magnitudes of the lag matrices of the HODMD method allows for some description of the preferred scales of the model, but it is nowhere near as clear as when using ERDMD to accomplish the same task.

Constructing the full Koopman one-step approximation  $\mathbf{K}_a$  as in Equation (3) allows us to find the affiliated Koopman spectrum as seen in the left side of Figure 4. We see that the affiliated characteristic polynomial  $p_a(z)$  is given explicitly by

$$p_a(z) = \det(\mathbf{K}_{149} + (\mathbf{K}_1 - z\mathbf{I})z^{148}).$$

If we look at the interior of the unit disc so that  $|z| < 1$ , then because of the strong separation in lags as seen in  $l_c$ , our innermost eigenvalues to leading order are found from the roots of  $\tilde{p}_{in,a}(z)$  where

$$\tilde{p}_{in,a}(z) = \det(\mathbf{K}_{149} + \mathbf{K}_1 z^{148})$$

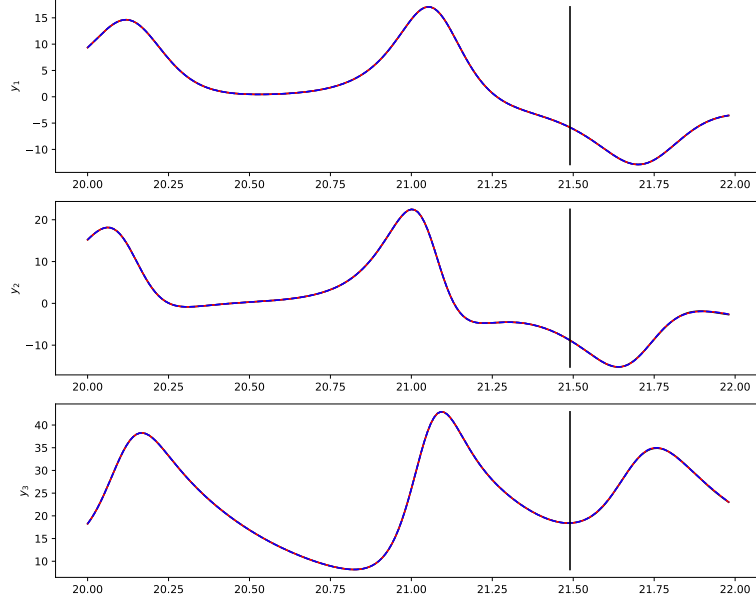
or the leading roots are found by finding the generalized eigenvalues  $\tilde{z} = z^{148}$  of the two matrix problem  $\mathbf{K}_{149} + \mathbf{K}_1 \tilde{z}$ .

Looking at the detail figure in Figure 4, we see that most of the features in the spectrum seen in the full spectrum on the left are present in the right. Thus, the damping in the dynamics comes almost entirely from the disparity in lag values. Further, we see that the full spectrum gets closer to the unit circle and even has two modes which just cross the unit circle. Otherwise, the reduced model well describes the damping modes, though we also see that the lag matrices  $\mathbf{K}_1$  and  $\mathbf{K}_{149}$  balance for more delicate dynamics along the unit circle.

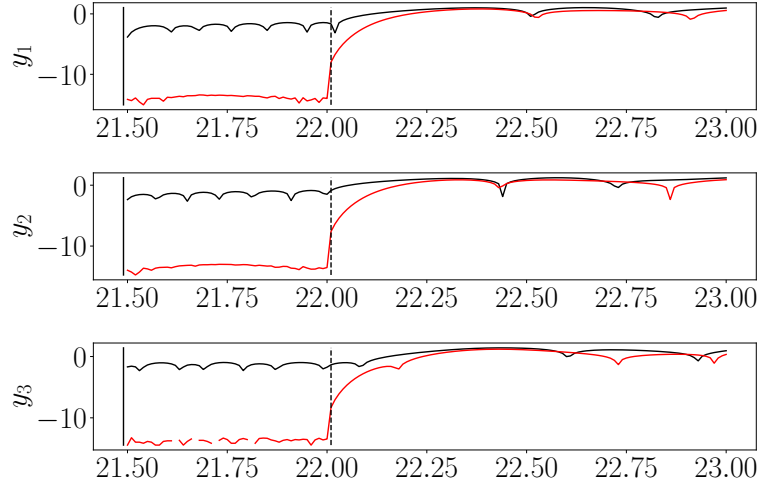
By way of contrast though, if we set the maximum lag  $d = 100$ , and build reconstructions for  $21 \leq t \leq 22$ , we find that the ERDMD results differ markedly in terms of the determined lags though ultimately not in terms of accuracy; see Figure 5. In this case, the ERDMD algorithm converges onto the lag choices

$$l_c = \{1, 15, 26, 35, 45, 48, 68, 73, 97, 99\}.$$

By choosing a lag horizon which does not fully capture the approximate period of oscillation in the dynamics, we need significantly more information to accurately reconstruct the data, though still nowhere as much as the full HODMD model. We also see though that the smaller choice of  $d$  causes



(a) Trajectories



(b) Error Comparison (semi-log scale)

Figure 2: Direct comparison of ERDMD and all lags HODMD model against the true trajectory for the Lorenz-63 system (a), and error across dimensions for the ERDMD and all lags HODMD method (b). The black line indicates the ERDMD result while the red indicates the all lags HODMD result. The solid vertical bar indicates the maximum lag choice of  $d = 150$ , while the dashed line indicates the end of the reconstruction interval and the beginning of the forecasting regime. The ERDMD algorithm converges to  $l_c = \{1, 149\}$ .

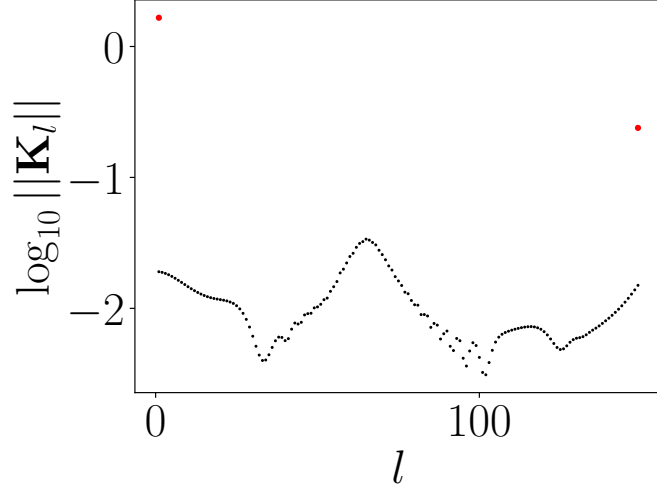


Figure 3: Comparison of the HODMD lagged matrix norms (black dots) and the ERDMD model (red dots) for the Lorenz-63 system with  $d = 150$ .

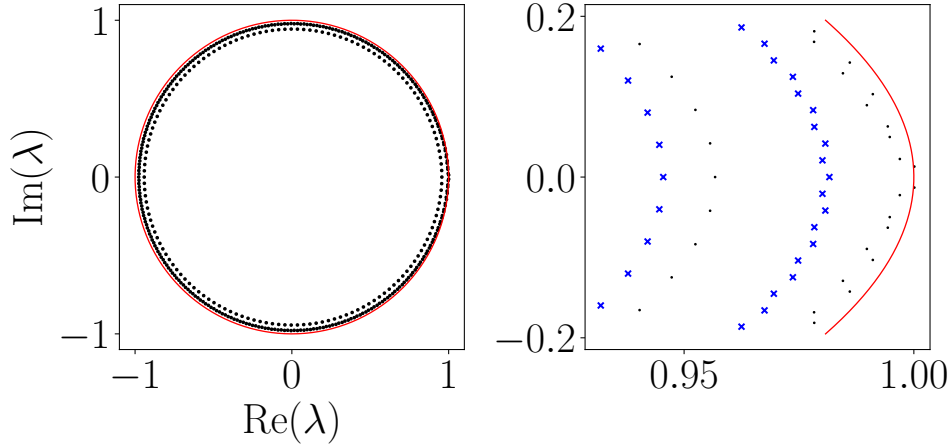


Figure 4: Spectrum of corresponding ERDMD Koopman operator for Lorenz-63 with  $d = 150$  on the left side, with a detail comparison to the roots of  $\tilde{p}_{in,a}(z)$  (blue crosses) on the right side near  $(1,0)$ . The ERDMD algorithm converges to  $l_c = \{1, 149\}$ . The solid/red line is the unit circle, provided for reference.

both models to essentially fail at providing any reasonable forecast, again reflecting the importance of good choices of  $d$  in the first place, a result which echoes the work in [7].

Comparing the models again as seen in Figure 6, we see that for  $d = 100$  we get a less minimal ERDMD model though with markedly larger matrix

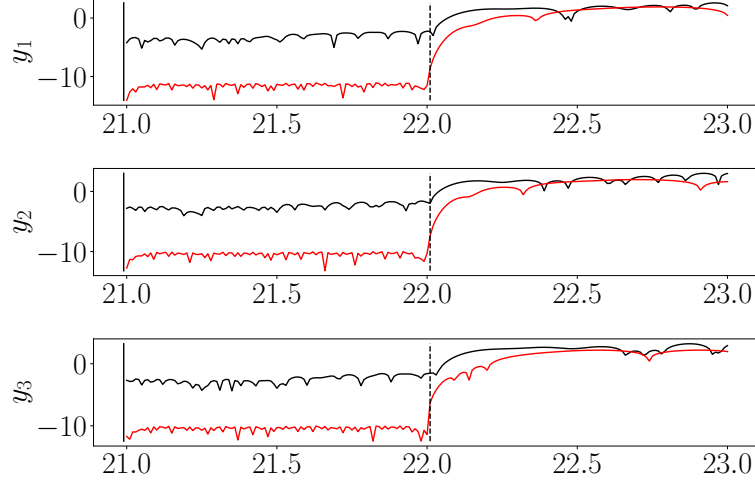


Figure 5: Comparison of ERDMD and all lags HODMD model against true trajectory for Lorenz-63 system. The ERDMD reconstruction is in solid black in the figure. The vertical bar indicates the maximum lag choice of  $d = 100$ . The ERDMD algorithm converges to  $l_c = \{1, 15, 26, 35, 45, 48, 68, 73, 97, 99\}$ .

norms for the longer lags, helping us to see that the model prioritizes long correlations in time in order to find accurate reconstructions. Likewise, the full HODMD model has some degree of spread in magnitudes, but it does not allow for ready analysis or description.

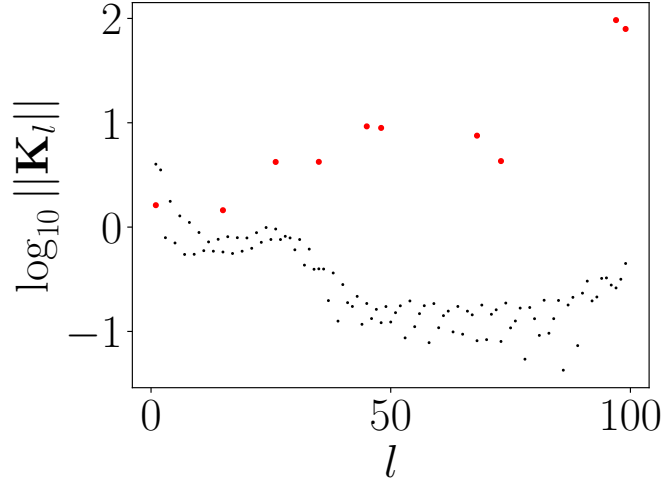


Figure 6: Comparison of the HODMD lagged matrix norms (black dots) and the ERDMD model (red dots) for the Lorenz-63 system with  $d = 100$ .

We can likewise find the spectrum of  $\mathbf{K}_a$  as seen in Figure 7. In this case, the larger spread of lags makes ready identification of positions in the spectrum to lag structure more difficult, though we see that The more

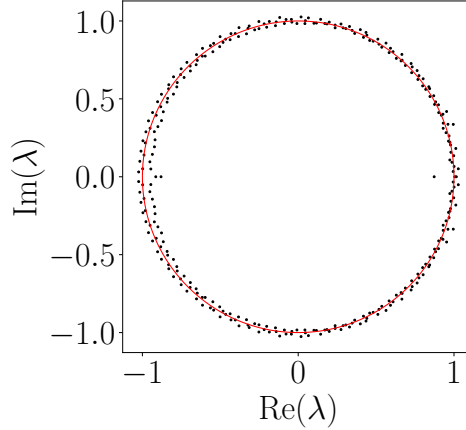


Figure 7: Spectrum of corresponding ERDMD Koopman operator for Lorenz-63 for  $d = 100$  on the left side, with a detail comparison to the reduced approximation (blue crosses) on the right side near  $(1,0)$ . The ERDMD algorithm converges to  $l_c = \{1, 15, 26, 35, 45, 48, 68, 73, 97, 99\}$ . The solid/red line is the unit circle, provided for comparison.

uniform spread in lag values makes any estimates of the spectrum using reduced models less useful beyond identifying the most strongly damped modes.

## Rossler Equation

To see how the ERDMD method works on problems with multiple scales, we now look at modeling dynamics coming from the Rossler system, given by the equations

$$\begin{aligned}\dot{y}_1 &= -y_2 - y_3 \\ \dot{y}_2 &= y_1 + ay_2 \\ \dot{y}_3 &= b + y_3(y_1 - c)\end{aligned}$$

where  $a = .1$ ,  $b = .1$ ,  $c = 14$ . To see the role the multiple time scales play in this problem, letting  $\epsilon = .1$ , then we see  $a = b = \epsilon$  and  $c = 1.4/\epsilon$ . Letting  $\tau = t/\epsilon$  and setting  $y_3 = \epsilon^2 \tilde{y}_3(t, \tau)$ , to leading order, in  $y_1$  and  $y_2$  we find

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = e^{\epsilon t/2} \begin{pmatrix} \cos(t) & -\sin(t) \\ \sin(t) & \cos(t) \end{pmatrix} \begin{pmatrix} y_{1,0} \\ y_{2,0} \end{pmatrix} + \mathcal{O}(\epsilon^2),$$

so that we have  $\mathcal{O}(1)$  planar oscillations complimented by slow growth away from the origin. Likewise, in  $\tilde{y}_3$  we find

$$\partial_\tau \tilde{y}_3 + \epsilon \partial_t \tilde{y}_3 = 1 - 1.4 \tilde{y}_3 + \epsilon y_1 \tilde{y}_3.$$

This then motivates the expansion

$$\tilde{y}_3(t, \tau) = c(t)e^{-1.4\tau} + \frac{1}{1.4} (1 - e^{-1.4\tau}) + \epsilon \tilde{y}_{3,1}(t, \tau) + \dots,$$

which, to remove secularities then gets us the equation for  $c(t)$  of the form

$$\frac{dc}{dt} = y_1(t) \left( c - \frac{1}{1.4} \right).$$

So we have

$$c(t) = \frac{1}{1.4} + \left( c(0) - \frac{1}{1.4} \right) e^{\int_0^t y_1(s) ds},$$

so that  $\tilde{y}_3$  stays small until the slow growth in  $y_1$  due to the  $e^{ct/2}$  term pushes the dynamics out of the plane.

We build reconstructions for  $25 \leq t \leq 40$ . To get accurate results, we chose  $d = 1000$ , for which choice the method converged to the set  $l_c$  with

$$l_c = \{1, 3, 170, 436, 553, 665, 988\}.$$

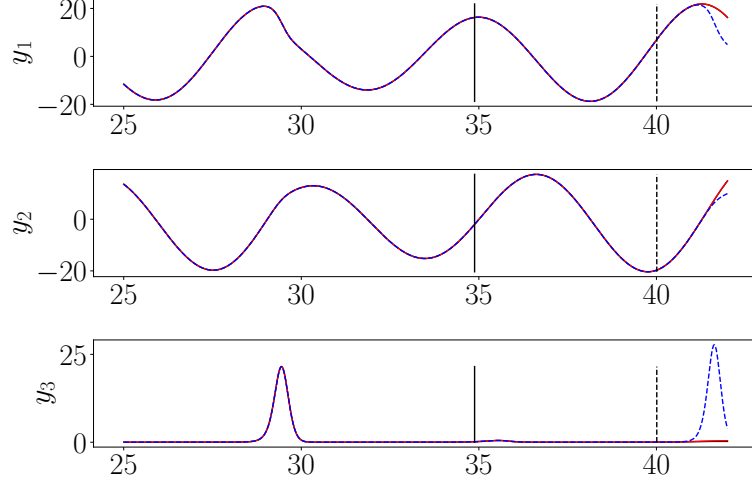
As can be seen in Figure 8, our accuracy relies on capturing a full departure from the fast through the slow manifold. This also explains the need for such a large choice of  $d$  relative to what was used for the Lorenz-63 system. Likewise, as with the Lorenz-63 system, the error in the ERDMD approximation is several orders of magnitude larger than for HODMD, though the practical difference is minimal. Neither model does well forecasting the next fast departarure from the planar oscillator reflecting an inherent difficult in learning slow/fast chaotic dynamics.

Comparing the HODMD and ERDMD models for the Rossler system, we see in Figure 9 somewhat peculiar results. Unlike for the Lorenz system, the norms of the lag matrices in the ERDMD model span ten orders of magnitude. Moreover, while  $\mathbf{K}_1$  and  $\mathbf{K}_3$  have relatively large norms, the remaining lag matrices are  $10^{-10}$  times smaller, so all of the long time lag matrices are categorically miniscule in comparison. So how can they have any meaningful impact within the model?

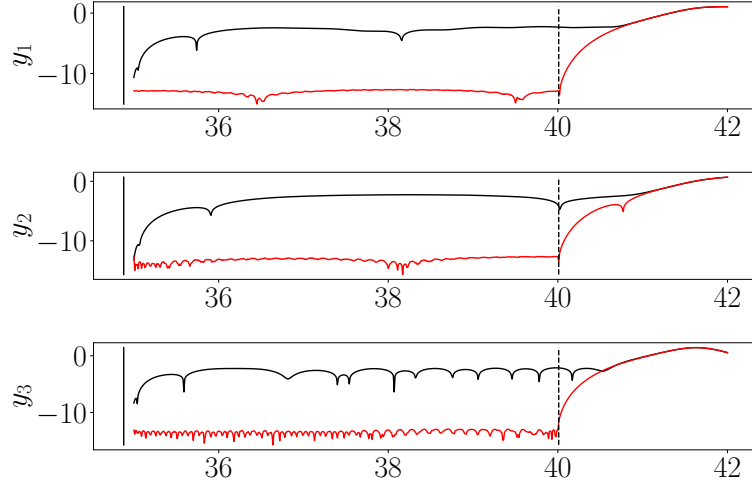
This question is answered in part by examining the spectrum, which will show us the spread in lag matrix norms is a result of the multiscale nature of the Rossler system. This of course appears in the wide spread of values of the optimal lags we see in  $l_c$  and the corresponding spread in norms of the affiliated lag matrices. Taking  $\epsilon = .1$ , we see that

$$\|\mathbf{K}_1\|, \|\mathbf{K}_3\| = \mathcal{O}(\epsilon^{-5}), \quad (10)$$

$$\|\mathbf{K}_{170}\|, \|\mathbf{K}_{436}\|, \|\mathbf{K}_{553}\|, \|\mathbf{K}_{665}\|, \|\mathbf{K}_{988}\| = \mathcal{O}(\epsilon^5). \quad (11)$$



(a) Trajectories



(b) Error Comparison

Figure 8: Comparison of ERDMD and all lags HODMD model against true trajectory for the Rossler system. The ERDMD reconstruction is in solid black in the figure. The vertical bar indicates the maximum lag choice of  $d = 1000$ . The ERDMD algorithm converges to  $l_c = \{1, 3, 170, 436, 553, 665, 988\}$ .

Thus, to look at substructure withing  $p_a(z)$  we should look at the respective balances

$$z^{988} \sim \epsilon^5, \quad z^{985} \sim \epsilon^{10}, \quad z^3 \sim \epsilon^{-5}, \quad (12)$$

so that for  $|z| < \epsilon^{10/985} \approx .977$ , we have

$$p_a(z) \approx \tilde{p}_{in,a}(z) = \det(z^{323} \mathbf{K}_{665} + \mathbf{K}_{988}). \quad (13)$$



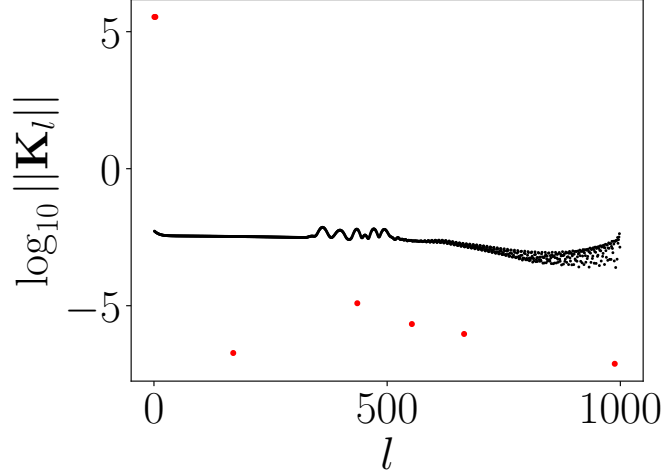


Figure 9: Comparison of the HODMD lagged matrix norms (black dots) and the ERDMD model (red dots) for the Rossler system.

The affiliated eigenvalues computed from this approximation correspond to the slowest scales in our ERDMD model. See Figure 10, right side detail for a comparison of the approximated eigenvalues to the actual values.

Likewise, if we look for eigenvalues outside the unit circle, we look to the fastest scales corresponding to the reduced polynomial

$$\tilde{p}_{out,a}(z) = \det(z^3 - \mathbf{K}_1 z^2 - \mathbf{K}_3)$$

We compare all of these regimes of the spectrum in Figure 10, in which we see most eigenvalues are close to or on the unit circle, though near  $(1, 0)$  as seen in the detail figure, we have a handful of modes outside. The largest in magnitude of these growing/unstable modes correspond to the fast scale roots coming from  $\tilde{p}_{out,a}(z)$ . Likewise, we see in the detail two rings of decaying modes which are reasonably well approximated by  $\tilde{p}_{in,a}(z)$ , which shows how the relatively minimal model generated by the ERDMD method allows for more ready identification, classification, and approximation of time scale related phenomena.

### Kuramoto–Sivashinsky Equation

To look at a more intricate and higher dimensional example, we now study the Kuramoto–Sivashinsky (KS) equation given by

$$u_t + u_{xx} + u_{xxx} + uu_x = 0, \quad u(x + 2L, t) = u(x, t).$$

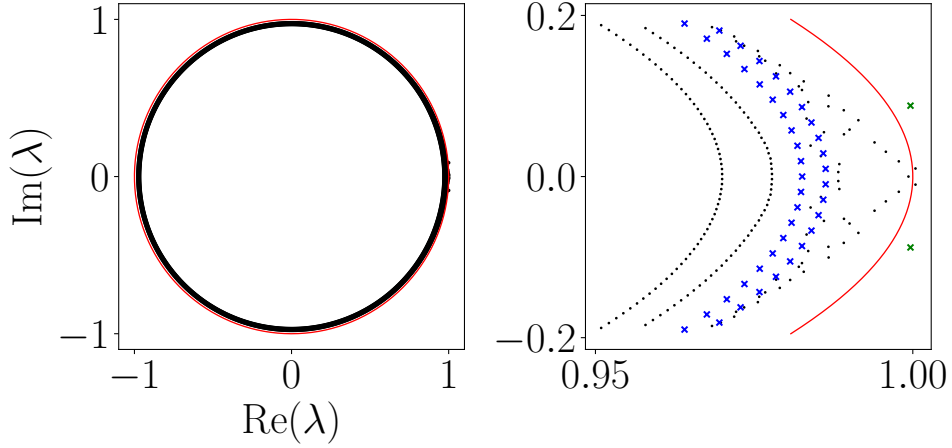


Figure 10: Spectrum of the corresponding ERDMD Koopman operator for the Rossler system on the left side, with a detail comparison to the roots of  $\tilde{p}_{in,a}(z)$  (blue crosses) and  $\tilde{p}_{out,a}(z)$  (green crosses) on the right side near  $(1,0)$ . The ERDMD algorithm converges to  $l_c = \{1, 3, 170, 436, 553, 665, 988\}$ . The solid/red line is the unit circle, provided for comparison.

See [18] for an extensive bibliography with regards to details and relevant proofs of facts used in this paper. Introducing the rescalings

$$\tilde{t} = \frac{t}{T}, \quad \tilde{x} = \frac{\pi}{L}x, \quad u = A\tilde{u},$$

and taking the balances

$$A = \frac{L}{\pi T}, \quad T = \left(\frac{L}{\pi}\right)^2,$$

we get the equivalent KS equation (dropping tildes for ease of reading)

$$u_t + u_{xx} + \nu u_{xxxx} + uu_x = 0, \quad \nu = \left(\frac{\pi}{L}\right)^2.$$

Looking at the linearized dispersion relationship  $\omega(k) = k^2 - \nu k^4$ , we see that the  $\nu$  parameter acts as a viscous damping term. Thus, as the system size  $L$  is increased, the effective viscosity is decreased, thereby allowing for more complex dynamics to emerge. As is now well known, for  $L$  sufficiently large, a fractional-dimensional-strange attractor forms which both produces intricate spatio-temporal dynamics while also allowing for a far simpler representation of said dynamics. It has been shown in many different works (see for example [19]) that  $L = 11$  generates a strange attractor with dimension between eight and nine, and that this is about the smallest value of  $L$

which is guaranteed to generate chaotic dynamics. We therefore set  $L = 11$  throughout the remainder of this section.

To study ERDMD on the KS equation, we use KS data numerically generated by a pseudo-spectral in space and fourth-order exponential-differencing Runge-Kutta in time method [20] of lines approach. For the pseudo-spectral method,  $K = 128$  total modes are used giving an effective spatial mesh width of  $2L/K = .172$ , while the time step for the Runge-Kutta scheme is set to  $\delta t = .25$ . After a burn in time of  $t_b = 10$ , we generated a simulation of length  $t_f = (L/\pi)^4 \approx 160$  to allow for nonlinear effects to fully manifest. This trajectory was then separated via a POD into space and time modes; see [21]. Taking  $N_s = 12$  modes captured 98.6% of the total energy.

To study the ERDMD method, we choose  $d = 200$ , which for  $dt = .25$  corresponds to a lag time of  $t = 60$ . With this choice, the ERDMD method finds  $l_c$  to be

$$l_c = \{1, 123, 141, 158\}.$$

The results for reconstruction can be seen in Figure 11, where we look at times  $10 \leq t \leq 66$ . As can be seen, the ERDMD does well, though we note that if we look for longer reconstructions, errors do start to appear more rapidly for the ERDMD method than the full HODMD method, and we are not able to push the ERDMD method to reproduce the entire reconstruction region from initial conditions. Thus, while the present results are promising, they represent an edge for our method that will need further tools to address; see [10] in this direction.

Repeating the HODMD and ERDMD model comparisons, we see in Figure 12 a situation similar to that seen in the Lorenz-63 system in so far as each matrix within our model is of roughly equal size and thus impact. Comparing to the Rossler system results, this shows there is no rapid transition layer or other distinguished time scale phenomena.

Nevertheless, we can still identify inner and outer approximations to eigenvalues affiliated with the ERDMD method. We see in Figure 13 the corresponding full spectrum affiliated with our method. Again, the strong separation of lags in  $l_c$  motivate looking at the roots of the reduced polynomial

$$\tilde{p}_{in,a}(z) = \det(z^{35}\mathbf{K}_{123} + z^{17}\mathbf{K}_{141} + \mathbf{K}_{158}).$$

Note, letting  $\tilde{z} = z^{17}$ , we can write

$$z^{35} = \tilde{z}^2 e^{\ln \tilde{z}/17} \approx \tilde{z}^2$$

so long as  $|z|$  is not too close to zero. We can then find the affiliated approximate companion matrix representation for  $\tilde{p}_a(z)$  in the form

$$\tilde{\mathbf{K}}_a = \begin{pmatrix} 0 & I \\ -\tilde{\mathbf{K}}_2 & -\tilde{\mathbf{K}}_1 \end{pmatrix}$$

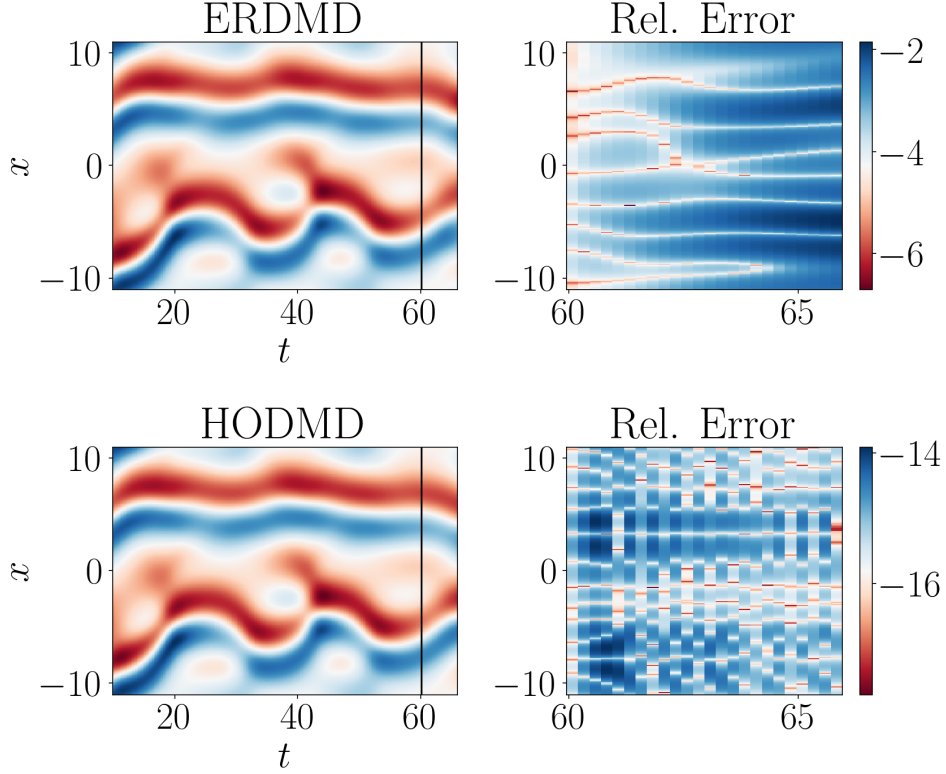


Figure 11: Comparison of ERDMD and all lags HODMD model against true trajectory for the KS equation. The vertical bar indicates the maximum lag choice of  $d = 200$ . The ERDMD algorithm converges to  $l_c = \{1, 123, 141, 158\}$ .

where

$$\tilde{\mathbf{K}}_1 = \mathbf{K}_{123}^{-1} \mathbf{K}_{141}, \quad \tilde{\mathbf{K}}_2 = \mathbf{K}_{123}^{-1} \mathbf{K}_{158}.$$

For this problem, for  $|z| > 1$ , we can also readily define and compute the roots of  $\tilde{p}_{out,a}(z)$  where

$$\tilde{p}_{out,a}(z) = \det(z - \mathbf{K}_1)$$

The full spectrum and detail comparison is seen in Figure 13, where we again can see the most damped modes come from the highest lags in our Koopman model. We likewise see that the most unstable mode of the full spectrum correspond to the largest magnitude eigenvalues of  $\mathbf{K}_1$ . Thus there is a clear separation in the Koopman spectrum across lag values, which is to say time scales in the dynamics.

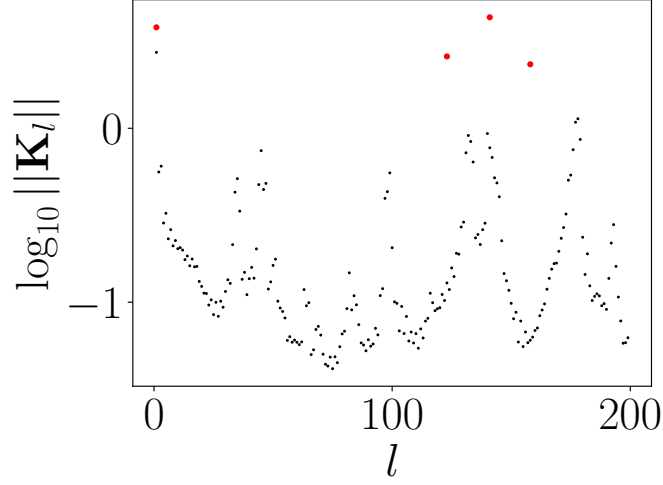


Figure 12: Comparison of the HODMD lagged matrix norms (black dots) and the ERDMD model (red dots) for the Kuramoto–Sivashinsky system.

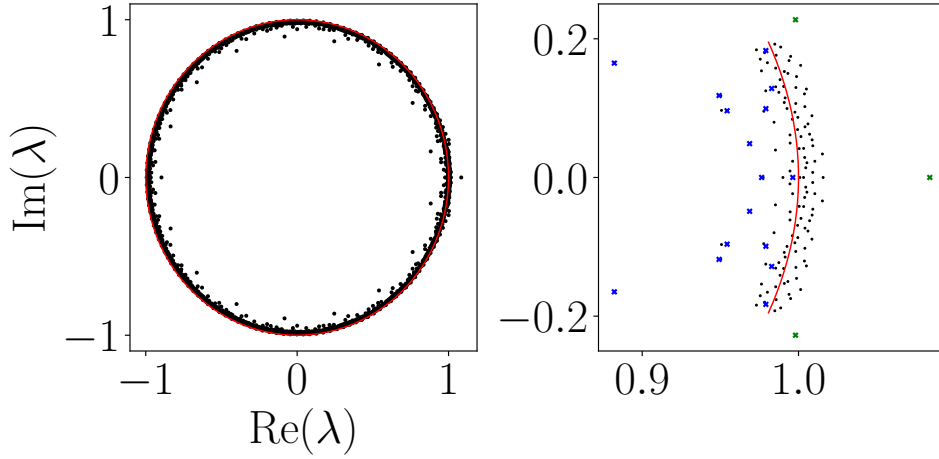


Figure 13: Spectrum of the corresponding ERDMD Koopman operator for the Kuramoto–Sivashinsky system on the left side, with a detail comparison to the roots of  $\tilde{p}_{in,a}(z)$  (blue crosses) and  $\tilde{p}_{out,a}(z)$  (green crosses) on the right side near  $(1,0)$ . The ERDMD algorithm converges to  $l_c = \{1, 123, 141, 158\}$ . The solid/red line is the unit circle, which is provided for reference.

## 4 Discussion

Discuss discuss discuss.

## Acknowledgements

## References

- [1] K. Taira, S.L. Brunton, S.T.M. Dawson, C.W. Rowley, T. Colonius, B.J. McKeon, O.T. Schmidt, S. Gordeyev, V. Theofilis, and L.S. Ukeiley. Modal analysis of fluid flows: An overview. *AIAA*, 55:4013–4041, 2017.
- [2] B. Lusch, J. N. Kutz, and S. L. Brunton. Deep learning for universal linear embeddings of nonlinear dynamics. *Nature Comm.*, 9:4950, 2018.
- [3] O. Azencot, N. B. Erichson, V. Lin, and M. Mahoney. Forecasting sequential data using consistent koopman autoencoders. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 475–485. PMLR, 2020.
- [4] D. J. Alford-Lago, C. W. Curtis, A. T. Ihler, and O. Issan. Deep learning enhanced dynamic mode decomposition. *Chaos*, 32:033116, 2022.
- [5] H. Arbabi and I. Mezic. Ergodic theory, dynamic mode decomposition, and computation of spectral properties of the Koopman operator. *SIAM Appl. Dyn. Sys.*, 16:2096–2126, 2017.
- [6] Soledad Le Clainche and José M. Vega. Higher order dynamic mode decomposition. *SIAM Journal on Applied Dynamical Systems*, 16(2):882–925, 2017.
- [7] Shaowu Pan and Karthik Duraisamy. On the structure of time-delay embedding in linear models of non-linear dynamical systems. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 30(7):073135, 07 2020.
- [8] Kathleen P. Champion, Steven L. Brunton, and J. Nathan Kutz. Discovery of nonlinear multiscale systems: Sampling strategies and embeddings. *SIAM Journal on Applied Dynamical Systems*, 18(1):312–333, 2019.
- [9] Mason Kamb, Eurika Kaiser, Steven L. Brunton, and J. Nathan Kutz. Time-delay observables for koopman: Theory and applications. *SIAM Journal on Applied Dynamical Systems*, 19(2):886–917, 2020.
- [10] Christopher W. Curtis, D. Jay Alford-Lago, Erik Bollt, and Andrew Tuma. Machine learning enhanced Hankel dynamic-mode decomposition. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 33(8):083133, 08 2023.

- [11] A. A. R. AlMomani, Jie Sun, and Erik M. Bollt. How entropic regression beats the outliers problem in nonlinear system identification. *Chaos*, 30:013107, 2020.
- [12] Thomas Schreiber. Measuring information transfer. *Phys. Rev. Lett.*, 85:461–464, Jul 2000.
- [13] Jie Sun, Dane Taylor, and Erik M. Bollt. Causal network inference by optimal causation entropy. *SIAM Journal on Applied Dynamical Systems*, 14(1):73–106, 2015.
- [14] Akaike Hirotogu. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19:716–723, 1974.
- [15] S.L. Brunton, B.W. Brunton, J.L. Proctor, , E. Kaiser, and J.N. Kutz. Chaos as an intermittently forced linear system. *Nature Comm.*, 8, 2017.
- [16] C. W. J. Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, 37(3):424–438, 1969.
- [17] Luca Faes, Giandomenico Nollo, and Alberto Porta. Information-based detection of nonlinear granger causality in multivariate processes via a nonuniform embedding technique. *Phys. Rev. E*, 83:051112, May 2011.
- [18] J.C. Robinson. *Infinite Dimensional Dynamical Systems*. Cambridge University Press, Cambridge, UK, 2001.
- [19] N. B. Budanur, P. Cvitanović, R. L. Davidchack, and E. Siminos. Reduction of  $so(2)$  symmetry for spatially extended dynamical systems. *Phys. Rev. Lett.*, 114:084102, Feb 2015.
- [20] AK Kassam and L.N. Trefethen. Fourth-order time-stepping for stiff PDEs. *SIAM J. Sci. Comp.*, 26:1214–1233, 2005.
- [21] G. Berkooz, P. Holmes, J.L. Lumley, and C.W. Rowley. *Turbulence, Coherent Structures, Dynamical Systems, and Symmetry*. Cambridge University Press, Cambridge, UK, 2012.