SMALL-DATA REDUCED ORDER MODELING OF CHAOTIC DYNAMICS THROUGH SYCO-AE: SYNTHETICALLY CONSTRAINED AUTOENCODERS

Andrey A. Popov[1,*] *& Renato Zanetti*[1]

[1]*Oden Institute for Computational Engineering & Sciences, The University of Texas at Austin, Austin, Texas, 78712*

[2]*Dept. of Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin, Austin, Texas, 78712*

*Address all correspondence to: Andrey A. Popov, Oden Institute for Computational Engineering & Sciences, The University of Texas at Austin, Austin, Texas, 78712, E-mail: andrey.a.popov@utexas.edu

Original Manuscript Submitted: mm/dd/yyyy; Final Draft Received: mm/dd/yyyy

Data-driven reduced order modeling of chaotic dynamics can result in systems that either dissipate or diverge catastrophically. Leveraging non-linear dimensionality reduction of autoencoders and the freedom of non-linear operator inference with neural-networks, we aim to solve this problem by imposing a synthetic constraint in the reduced order space. The synthetic constraint allows our reduced order model both the freedom to remain fully non-linear and highly unstable while preventing divergence. We illustrate the methodology with the classical 40-variable Lorenz '96 equations and with a more realistic fluid flow problem—the quasi-geostrophic equations—showing that our methodology is capable of producing medium-to-long range forecasts with lower error using less data than other non-linear methods.

KEY WORDS: *reduced order modeling, autoencoders, chaos, non-intrusive, non-linear dimensionality reduction*

1. INTRODUCTION

The use of machine learning methods in many scientific disciplines is hampered by the availability of data (Kitchin and Lauriault, 2015), resulting in the small-data problem. Knowledge-guided (Karpatne et al., 2022) machine learning (Aggarwal et al., 2018; Goodfellow et al., 2016) aims to, in part, solve this problem by augmenting data driven methods with *a priori* knowledge about the true system behavior. This prior knowledge can act as a regularizer, which, for instance, can restrict neural network outputs to remain physically consistent.

Chaotic systems—commonplace in numerical weather prediction and data assimilation applications (Asch et al., 2016; Kalnay, 2003; Reich and Cotter, 2015)—have the peculiar property of both magnifying small perturbations and damping them at the same time (Guckenheimer and Holmes, 2013; Strogatz, 2018). Because of this, the dynamics of chaotic systems are hard to predict, being linearly unstable on average, while the limit set of the dynamics, the attractor,

2689–3967/20/$35.00 © 2020 by Begell House, Inc. www.begellhouse.com

remains compact (Farmer et al., 1983), resulting in quasi-periodic behavior. In practice all the points of the attractor cannot be known: it is possible that there is very limited data about the attractor both temporally and spatially. These attractors often possess complicated topological properties (Rand, 1978) and are difficult to work with for high dimensional systems,.

Reduced order modeling (ROM) (Benner et al., 2017) combines the ideas of dimensionality reduction and the idea of operator inference (Brunton and Kutz, 2022) in order to build efficient models of dynamical systems. In the world of machine learning dimensionality reduction is frequently performed through the use of autoencoders (Goodfellow et al., 2016; Popov et al., 2022; Weng, 2018), which are the focus of this work.

In this work, we make the simple assumption that we can construct an autoencoder and reduced order dynamics such that the chaotic attractor can be embedded into a simple-to-define set in reduced order space. This can be achieved by imposing a *synthetic constraint* on both the autoencoder and the reduced order dynamics. If the synthetic constraint defines a compact set, we can guarantee that the reduced order dynamics stay bounded while simultaneously having all the non-linear freedom that generalized function approximators allow. The synthetic constraint therefore becomes a *simple and known* transformation of some *unknown* constraint of the full model, and, acts as a proxy for our knowledge about the behavior of chaotic systems.

This work aims to build reduced order models of chaotic systems by combining autoencoders, fully non-linear operator inference, and synthetic constraints. We call this framework synthetically constrained autoencoders, shortened to SyCo-AE. We provide a theoretical justification for the derivation of the SyCo-AE through the construction of the strong and weak preservation properties. We show how an ideal reduced order model satisfies the strong preservation property, and how the SyCo-AE is built explicitly to preserve the weak preservation property. We additionally provide a novel numerical method for training this reduced order model, by embedding the solution of a constrained differential equation into the cost function.

We apply the SyCo-AE framework first to the Lorenz '96 equations (Lorenz, 1996), which is a common medium-scale highly-chaotic test problem, and second to the quasi-geostrophic equations (Ferguson, 2008; Foster et al., 2013; Greatbatch and Nadiga, 2000; Majda and Wang, 2006). Our results show, both qualitatively and quantitatively, that the SyCo-AE framework is capable of producing a reduced order model that can produce reliable medium-range forecasts of chaotic dynamics.

Of note, and a large motivation, though not a focus, of this work is providing a framework for constructing reduced order models for numerical weather prediction and data assimilation (Asch et al., 2016; Kalnay, 2003; Reich and Cotter, 2015), where chaotic models of the physical processes are commonplace. In this context, reduced order models are frequently not used in isolation, but as parts of multilevel and multifidelity frameworks (Chada et al., 2020; Hoel et al., 2019; Popov and Sandu, 2023).

This paper is organized as follows. Section 2 presents some background on the non-linear dimensionality reduction problem, and previous attempts at linear and non-linear non-intrusive reduced order models. Section 3 presents the synthetically constrained autoencoder framework. Section 4 presents numerical experiments with the Lorenz '96 and Quasi-geostrophic equations. Finally section 5 has some concluding remarks.

2. BACKGROUND

Reduced order modeling combines two concepts: *dimensionality reduction*, and *operator inference*. The former primarily deals with preserving *spatial* features of the dynamical system through a compressed representation while the latter focuses on preserving the *temporal* features of the dynamical system.

We now describe dimensionality reduction from the point of view of autoencoders. The autoencoder,

$$\begin{aligned} u &= \theta(x), \\ x \approx \widetilde{x} &= \phi(u), \end{aligned} \tag{1}$$

aims to take information given by the high-dimensional state x, and distill it down to some useful lower-dimensional representation u, then back out again into a reconstruction $\widetilde{x}$. More formally, assume that x is an element of the full order space $\mathbb{X}$ which is a subset of n-dimensional Euclidean space $\mathbb{X} \subset \mathbb{R}^n$. The encoder is a function $\theta : \mathbb{R}^n \to \mathbb{R}^r$, with the reduced order space defined by the image of the full order space under the encoder, $\theta(\mathbb{X}) = \mathbb{U}$, which itself is a subset of r-dimensional Euclidean space, $\mathbb{U} \subset \mathbb{R}^r$. The decoder is a function $\phi : \mathbb{R}^r \to \mathbb{R}^n$, with the reconstruction space defined by the image of the reduced order space under the decoder $\phi(\mathbb{U}) = \widetilde{\mathbb{X}}$.

The encoder and decoder in eq. (1) are functions that are required to have additional properties for dimensionality reduction to be valid. A necessary, but not sufficient condition for valid dimensionality reduction on the encoder-decoder pair is that of right-invertibility (Popov et al., 2022),

$$\theta(\phi(\theta(x))) = \theta(x), \quad \forall x \in \mathbb{X}, \tag{2}$$

over all the data. This property makes sure that there is no loss of information from the reduced order representation u in the reconstruction $\widetilde{x}$, as it would encode into the exact same u.

The true state that we are trying to model is assumed to come from some continuous dynamical system defined by the differential equation,

$$\frac{\mathrm{d}x}{\mathrm{d}t} = F(x), \tag{3}$$

with F representing the (possibly highly non-linear) dynamics. The goal of this work is to approximate the *full order model* eq. (3) in the reduced space defined by the autoencoder eq. (1). This model is known as a *reduced order model*, though the term is broadly applied to a wide range of methods Benner et al. (2017) not all of which are concerned with time-dependent differential equations.

In *intrusive* approaches to finding the reduced order model of the full order model eq. (3) is known explicitly and is used in order to take advantage of all available information. However when eq. (3) is not known, or when our knowledge about it is severely deficient, intrusive methods cannot be relied upon. In this work we focus on *non-intrusive* methods that do not have access to the full order model, and must rely only on data.

Given the state of the dynamics at time index i in the reduced space, denoted by u_i, the goal is to find the state of the dynamics at time index $i + 1$, denoted by u_{i+1}. Conceptually u_{i+1} can be approximated in two major ways. The first is that of *flow maps* (Qin et al., 2019), where a simple function that maps one to the other is learned,

$$u_{i+1} \approx \mathcal{F}(u_i). \tag{4}$$

The flow map approach eq. (4) works best when the data are spaces a constant time-step $\Delta t = t_{i+1} - t_i$ apart, as incorporating the time-step into the flow map operator can fail to generalize to time-steps outside of the training data.

While conceptually simple, the flow map approach eq. (4) approach does not lend well to continuous dynamical systems, as the attractors for discrete and continuous dynamics do not behave in the same manner (Guckenheimer and Holmes, 2013). The alternate approach, which we utilize in this paper is that of *operator inference*, whereby the action of the continuous time dynamics in the reduced space is learned explicitly,

$$\frac{\mathrm{d}u}{\mathrm{d}t} = f(u), \quad u(t_i) = u_i, \quad t \in [t_i, t_{i+1}], \tag{5}$$

where f is some (potentially highly non-linear) function defining the dynamics in the reduced space, t is the time of the full order dynamics, and u_i is the initial condition. The solution of the initial value problem eq. (5) can be performed with a wide array of algorithms (Hairer et al., 1991). Another strength of the operator inference approach is that the initial value problem eq. (5) can be solved to any arbitrary final time, thus allowing for the forecast to be either interpolated and extrapolated in time.

The most straight-forward autoencoder-based approach to reduced order modeling of continuous dynamics take a fully non-linear autoencoder-based dimensionality reduction method coupled to a fully neural-network-based operator inference step,

$$\begin{gathered} u = \theta(x), \quad \widetilde{x} = \phi(u), \\ \frac{\mathrm{d}u}{\mathrm{d}t} = f(u), \end{gathered} \tag{6}$$

where θ and ϕ are the autoencoder eq. (1), and f is a full non-linear approximation of the f found in eq. (5). In this work we augment the approach in eq. (6).

Another issue with the flow map approach eq. (4) is in control applications (Khalil, 2015; Stengel, 1994), as it would only be possible to apply a control every Δt steps, while the operator inference approach eq. (5) is able to support a continuous control signal through the trivial addition,

$$\frac{\mathrm{d}u}{\mathrm{d}t} = f(u) + \left.\frac{\mathrm{d}\theta}{\mathrm{d}x}\right|_{\phi(u)} f_c(z), \tag{7}$$

where f_c is the control function and z is the control signal. The extension to control is not explored in this work.

2.1 Other Non-Intrusive Methods

We now discuss some previous non-intrusive methods, that this work compares against.

One method for performing non-intrusive reduced order modeling is dynamic mode decomposition (DMD) (Brunton and Kutz, 2022; Kutz et al., 2016). Given a set of data points, $\{(x_i, x_{i+1})\}_i$, DMD finds the best rank-r linear operator $\mathbf{A}$ that transports the data from time index i to time index $i+1$, given by $x_{i+1} \approx \mathbf{A}x_i$. Taking the eigendecomposition of $\mathbf{A}$, given by $\mathbf{A}\boldsymbol{\Phi} = \boldsymbol{\Phi}\boldsymbol{\Lambda}$, we can define the reduced order model in the following way,

$$\begin{gathered} u = \theta(x) = \boldsymbol{\Phi}^{\dagger}x, \quad \widetilde{x} = \phi(u) = \boldsymbol{\Phi}u, \\ \frac{\mathrm{d}u}{\mathrm{d}t} = \boldsymbol{\Omega}u, \quad \boldsymbol{\Omega} = \frac{1}{\Delta t}\log\boldsymbol{\Lambda}, \end{gathered} \tag{8}$$

where $\boldsymbol{\Phi}$ is our linear decoder, its pseudo-inverse, $\boldsymbol{\Phi}^{\dagger}$ is the encoder, and $\boldsymbol{\Omega}$ is the linear operator defining the time dynamics. The initial value problem eq. (8) has a linear analytic solution, which is used in this work.

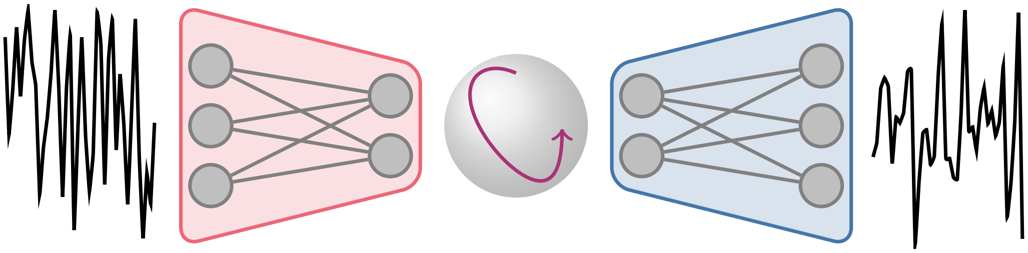


FIG. 1: A visual representation of the SyCo-AE framework. From left to right: full order model eq. (3) data is encoded into the reduced order space, the reduced order model eq. (5) is evolved on the sphere S^{r-1}, and then decoded out by the decoder, into the reconstruction. Both left and right curves in this figure are real data from the Lorenz ’96 model described in section 4.

The advantage of linear methods is that they require few data to converge to a useful solution. In general, the amount of data required to use a linear method is on the order of the reduced dimension r and significantly less than the dimension of the full order model n.

However linear methods fail to produce useful results for highly non-linear systems and are incapable of modeling chaotic dynamics. As such, non-linear reduced order modeling methods are required.

Moving away from linear methods, a recent more-than-linear approach is that of quadratic manifolds (Geelen et al., 2023), which requires $\mathcal{O}(r^2)$ data points to construct,

$$\begin{gathered} u = \theta(x) = \mathbf{\Phi}^{\mathsf{T}}(x - \bar{x}), \quad \widetilde{x} = \phi(u) = \bar{x} + \mathbf{\Phi} u + u^{\mathsf{T}}\,\overline{\mathbf{\Phi}} u, \\ \frac{\mathrm{d}u}{\mathrm{d}t} = a + \mathbf{B}u + u^{\mathsf{T}}\mathcal{C}u, \end{gathered} \tag{9}$$

where $\bar{x}$ can be taken to be the mean-field of the data, $\mathbf{\Phi}$ is a matrix constructed by proper orthogonal decomposition, and $\overline{\mathbf{\Phi}}$ is a 3-tensor defining the quadratic correction term. The terms a, $\mathbf{B}$ and $\mathcal{C}$ represent the quadratic approximation of the reduced order model. The quadratic manifolds approach first constructs the dimensionality reduction, and then constructs the dynamics from the data approximating the time derivative of u through finite difference. Sparsity in the linear operators and 3-tensors can be enforced in a way similar to the sparse identification of non-linear dynamics (SINDy) method (Brunton et al., 2016), though this type of regularization is not explored in this work.

In both the classic approaches above, the encoder is an affine transformation, meaning that the image of $\mathbb{R}^n$ under the encoder is $\theta(\mathbb{R}^n) = \mathbb{R}^r$, and not a compact superset of the reduced order space $\mathbb{U}$. Thus, the encoder is not restricted to producing results in $\mathbb{U}$. This means that they can produce results that are not meaningful to the problem at hand.

3. SYNTHETICALLY CONSTRAINED AUTOENCODER

If we know that the system that we wish to model is chaotic, then, given an encoder that preserves compactness, the reduced order dynamics have to always remain on some compact set. If they do not then our constructed reduced order model did not take advantage of all our available knowledge, which is a violation of our current best understanding of scientific reasoning (Jaynes, 2003). This type of violation is often done deliberately for lack of a better solution. Our task, therefore, is to somehow include this knowledge into our reduced order model construction.

A straightforward consequence of this reasoning is that the construction of the autoencoder eq. (1) cannot be performed independently of the construction of the reduced order dynamics eq. (5). We have to not only ensure that the reduced order dynamics produce outputs that are in accordance with the reduced order space induced by the encoder, but also ensure the complement: that the encoder produces outputs that are in accordance to the outputs produced by the reduced order dynamics.

We now formalize the discussion above by defining a few properties, and their straightforward consequences.

Definition 3.1. Ideally the dynamics in eq. (5) evolve in a way such that the final state is always in the reduced order space $u_{i+1} \in \mathbb{U}$, which we term this the *strong preservation condition*.

We now outline how the strong preservation condition in definition 3.1 could be satisfied when the attractor is known.

Theorem 1. *When all the possible data points of the attractor* $\mathbb{X}$ *are fully known, and* θ, ϕ *and* f *in the standard autoencoder reduced order model eq.* (6) *are allowed to have an arbitrary amount of degrees of freedom, then the strong preservation condition in definition 3.1 is preservable.*

Proof. With arbitrary many degrees of freedom, the discrepancy of the propagation u_{i+1} and the dimensionality reduced representation of the truth, $\theta(x_{i+1})$, can be zero almost surely for all $x \in \mathbb{X}$, assuming that the autoencoder eq. (1) and reduced order dynamics eq. (5) found simultaneously. □

In practice the attractor, $\mathbb{X}$, is not fully known, and in the small-data problem, the known points might be a biased representation thereof. Therefore, the encoder must practically accept all points in $\mathbb{R}^n$ as its input. The set defined by the image of $\mathbb{R}^n$ under the encoder, $\theta(\mathbb{R}^n) = \widehat{\mathbb{U}}$ we call the *total reduced order space*, as it fully covers all possible inputs to the encoder.

Definition 3.2. If the reduced order dynamics eq. (5) evolve in a way such that the final state is always in this total reduced order space $u_{i+1} \in \widehat{\mathbb{U}}$, we term this the *weak preservation condition*.

We now motivate our subsequent discussion with the following trivial result.

Theorem 2. *A shallow neural network with one hidden layer,*

$$\nu(x) = \mathbf{A}_2\, \sigma(\mathbf{A}_1 x + \mathbf{b}_1) + \mathbf{b}_2, \tag{10}$$

with continuous activation function σ*, maps compact sets to compact sets.*

Proof. Compactness is preserved by continuous functions, thus the composition of an affine, continuous, and another affine function is itself continuous. □

We can attempt to learn $\widehat{\mathbb{U}}$ for a given encoder, and force that the dynamics lie on this set. This is inadequate for one simple reason: there is no guarantee that $\widehat{\mathbb{U}}$ is compact, thus being a poorly positioned solution to the problem of modeling chaotic dynamics.

In this work we aim to solve this problem by turning it on its head: instead of learning $\widehat{\mathbb{U}}$ we instead *synthetically assign* $\widehat{\mathbb{U}}$ to be some compact lower-dimensional manifold embedded in $\mathbb{R}^r$. We can then restrict both the encoder eq. (1) and dynamics eq. (5) to map onto and evolve on this manifold. This type of restriction can be performed through a simple algebraic constraint. We first take the autoencoder reduced order model eq. (6), and augment it,

$$\boxed{\begin{aligned} &u = \theta(x), \quad \widetilde{x} = \phi(u), \\ &\frac{\mathrm{d}u}{\mathrm{d}t} = f(u), \quad 0 = g(u), \end{aligned}} \tag{11}$$

where the addition of the synthetic constraint function $g : \mathbb{R}^r \to \mathbb{R}^s$ with $s \leq r$ has an effect on both the encoder θ and on the dynamics f. The resulting dynamics define a constrained ordinary differential equation (Ascher and Petzold, 1998). This synthetically constrained autoencoder (SyCo-AE) is described visually by fig. 1.

The hope is that synthetically constraining the dynamics to a known low dimensional set would act as a source of additional knowledge, in the Bayesian sense of the word Jaynes (2003). In other words it is information that is potentially an inherent property of the system from which the data are taken, but is not representable by a limited data set. This knowledge should guide the neural networks to be a good approximation of the underlying dynamics with significantly fewer data. We can also alternatively think about implicitly defining a constraint in the full space. We conjecture:

Conjecture 3.
We approximate a complex, not-yet-known, constraint on the full space, $0 = G(x) \approx G(\phi(u))$ for all reduced order states u that satisfy our synthetic constraint $0 = g(u)$.

In other words, by introducing a *known and simple* constraint on the reduced order dynamics, we learn some *unknown and complex* constraint of the full order dynamics. Note that the range of G and g might not have the same dimension, meaning that the optimal choice of g should somehow be informed by the dimension of the range of the unknown G.

One simple way to solve enforce the constraint g in eq. (11) for both the encoded and dynamics is through projection. Given the constraint g, we can pose the projection operator as minimizing a cost,

$$\Pi(u) = \arg\min_{\widehat{u}} \|\widehat{u} - u\|_2^2, \qquad \text{such that} \quad 0 = g(\widehat{u}), \tag{12}$$

which for certain choices of g can have a closed form solution. Given an arbitrary encoder $\widehat{\theta}$, the projected autoencoder is given by,

$$\theta(x) = \Pi\Big(\widehat{\theta}(x)\Big). \tag{13}$$

In this work for the solution of all continuous dynamical systems, in the interest of speed, we utilize the projected adaptive time-step explicit trapezoidal rule,

$$\begin{aligned} &k_1 = h_j\, f(u_j), \quad k_2 = h_j\, f(u_j + k_1), \\ &u_{j+1} = \Pi\left[u_j + \frac{1}{2}(k_1 + k_2)\right], \quad \widetilde{u}_{j+1} = \Pi[u_j + k_1], \end{aligned} \tag{14}$$

where h_j is the internal time-step of the solver, u_{j+1} is the next state at time $t_j + h$, $\widetilde{u}_{j+1}$ is the solution of the 'embedded' method used to adaptively determine h_j through the canonical method described in (Hairer et al., 1991), and the function Π from eq. (12) projects the solutions onto the manifold defined by g in eq. (11) or is the identity function for all other methods. The ideas for this method are a simple combination of ideas found in (Ascher and Petzold, 1998; Hairer et al., 2006, 1991; Hairer and Wanner, 1996).

Remark 4 (Index-2 DAE formulation)**.** *The constrained dynamics defined in eq.* (11) *can be posed (Ascher and Petzold, 1998) and solved (Hairer et al., 2006; Hairer and Wanner, 1996) as an index-2 differential algebraic equation. This formulation is not explored in this work.*

We now turn our attention to the choice of $\widehat{\mathbb{U}}$, implicitly defined by the constraint g in eq. (11). We list a few properties that are nice to have: (i) $\widehat{\mathbb{U}}$ is a compact (and connected) subset of $\mathbb{R}^r$, (ii) its topological dimension (Heinonen et al., 2001) is as close to r as possible, and (iii) the projection Π in eq. (12) has a closed form solution. One simple candidate for such a set is the sphere S^{r-1}, with corresponding constraint and projection of,

$$g(u) = ||u||_2 - 1, \quad \Pi(u) = \frac{u}{||u||_2}, \tag{15}$$

that has topological dimension of $r - 1$. We make use of the spherical constraint eq. (15) for the remainder of this work, and table the discussion of other possible constraints for future work.

Remark 5. *Other possible choices of g include constraining the system to a lower dimensional open manifold such as,*

$$g(u) = 1 - \sum_i u_i, \tag{16}$$

which does not define a compact manifold. Another choice is constraining to the surface of an r-torus, recursively defined as

$$\begin{aligned} g_r(u) &= u_r^2 + g_{r-1}^2(u) - d_{r-1}^2 \\ g_s(u) &= \sqrt{u_s^2 + u_s^2} - d_{s-1}, \quad s = r-1, \ldots, 2 \\ g_1(u) &= u_1 \end{aligned} \tag{17}$$

which is a torus as long as the radii $\{\mathrm{d}\}$ satisfy $\mathrm{d}_s > \mathrm{d}_{s-1}$ for $s = r - 1, \ldots, 1$. As far as the authors are aware, there is no simple closed form projection operator for the r-torus, thus this choice would involve the solution to a non-linear optimization problem. In the authors' experience this approach significantly slows down the training with no apparent benefit to decrease in error.

A third choice is to simply define multiple spherical constraints on subsets of the states of u, though this choice suffers from a combinatoric explosion of choice, for what the authors suspect is marginal computational benefit.

We organize the data into trajectories, with each trajectory, $\{X_k\}_{k=0}^K$, consisting of data at times $\{t_k\}_{k=0}^K$. In order to effectively learn the dynamics we explicitly *roll out* (Uy et al., 2022) eq. (11), by explicitly solving the constrained initial value problem during training. The degree to which roll out is performed is determined by the length of the trajectory K, thus we term K the roll out parameter. We write $\mathcal{M}_{t_k \to t_{k+1}}(u_k)$ for the solution of the constrained initial value problem eq. (11) with an algorithm such as eq. (14). Note that each trajectory can be of variable length and with variable inter-trajectory time-step, however this idea is not explored in this work.

The cost function, therefore has to take into account four distinct parts: (i) the autoencoder error eq. (1), (ii) the right-invertability condition eq. (2), (iii) the error of the dynamics eq. (11) in the full space (through the decoder), (iv) the error of the dynamics eq. (11) in the reduced order space, required by theorem 1. The resulting cost function,

$$\begin{aligned}\ell_i(\{t_k\},\{X_k\})|_W = \quad & \sum_{k=0}^{K} \underbrace{||X_k - \phi(\theta(X_k))||_2^2}_{\text{Autoencoder error}} + \omega \sum_{k=0}^{K} \underbrace{||\theta(X_k) - \theta(\phi(\theta(X_k)))||_2^2}_{\text{Right-inverse error}} \\ + \quad & \sum_{k=1}^{K} e^{-2\lambda(t_k - t_0)} \underbrace{||X_k - \phi(\mathcal{M}_{t_0\to t_k}(\theta(X_0)))||_2^2}_{\text{Full space dynamics error}} \qquad , \\ + \upsilon & \sum_{k=1}^{K} e^{-2\lambda(t_k - t_0)} \underbrace{||\theta(X_k) - \mathcal{M}_{t_0\to t_k}(\theta(X_0))||_2^2}_{\text{Reduced space dynamics error}}\end{aligned} \tag{18}$$

is implicitly defined in terms of the neural network weights W. The constant ω manipulated the the right inverse error enabling the autoencoder to be weakly right-invertible eq. (2). A large choice, $\omega = 10^2$, has been shown (Popov and Sandu, 2023) to significantly improve the performance of autoencoder-based reduced order models. This value of ω is used in this work. For chaotic systems, the dominant error of the system grows exponentially on average, thus the errors at different times forward into the model have to be scaled accordingly. This accumulation of the error is regulated by λ which can be thought of as the approximation of the largest Lyapunov exponent (LLE) (Parker and Chua, 2012) for the given dynamics. For a system that is not chaotic, and which can be exactly reconstructed by a reduced order model of dimension r, the LLE can be set to $\lambda = 0$, meaning that we assume there is no accumulation of error forward in time. This might not strictly be the case for most interesting systems. For chaotic systems, λ is either known, or can be tuned as a hyperparameter. Finally, υ is a parameter that scales the loss in the reduced order space with respect to the autoencoder loss.

Taking a collection of I trajectories, $\{\{X_k\}_i\}_{i=1}^{I}$, the full cost function is the expected value with respect to all the trajectories,

$$L(W) = \mathbb{E}[\ell_i(\{t_k\}_i, \{X_k\}_i)|_W]. \tag{19}$$

Note that the cost eq. (18) has the embedded time integration algorithm eq. (14), reminiscent of the idea of neural differential equations (Chen et al., 2018) and neural manifold differential equations (Lou et al., 2020), thus the solution to this cost function through the help of automatic differentiation could be characterized as differentiable programming (Baydin et al., 2018).

Remark 6 (Largest Lyapunov Exponent)**.** *The optimal LLE of the reduced order model, for a given error metric, is not necessarily the LLE of the full order model. In the authors' experience, having this value be a hyper parameter has not shown significant decrease in error. From a model design perspective, if the LLE is known for the full order model, it should be used to construct the reduced order model.*

4. NUMERICAL EXPERIMENTS

We present two numerical experiments, one with a small 40-variable problem and another with a more realistic fluid flow problem.

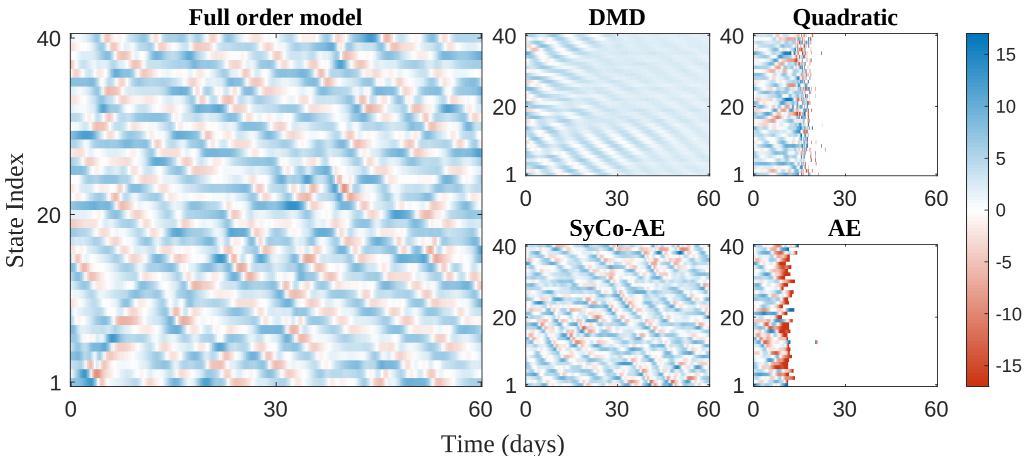


FIG. 2: Qualitative representation of the forecasting accuracy of the various ROM methods as compared to the full order Lorenz ’96 model. The same initial condition is taken for all models and propagated for 60 days. The x-axis is the time in days, and the y-axis is the state-space index of each variable.

4.1 Lorenz ’96 Equations

For our numerical test case we take the Lorenz ’96 equations (Lorenz, 1996),

$$\frac{\mathrm{d}x_j}{\mathrm{d}t} = -x_{j-1}(x_{j-2} - x_{j+1}) - x_j + F, \quad j \in [1, \dots, 40], \tag{20}$$

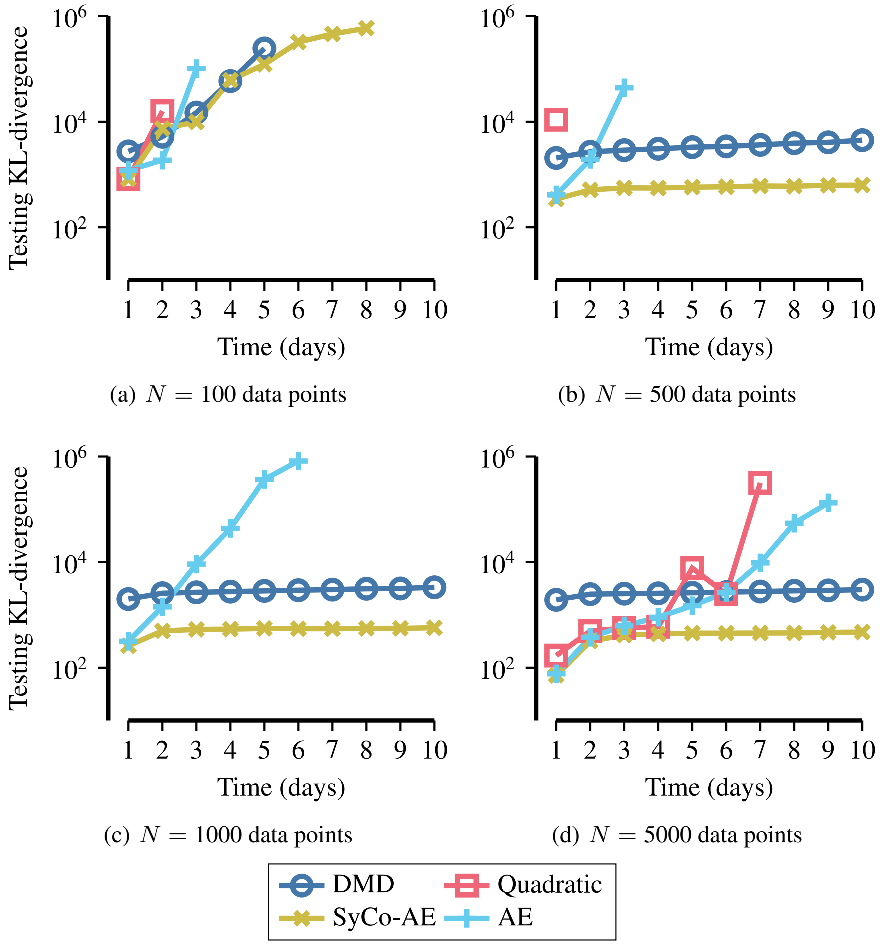
with cyclic boundary conditions in j, and external force $F = 8$. The implementation of the Lorenz ’96 equations used in this work is taken from the ODE Test Problems (Roberts et al., 2019) package.

The Lorenz ’96 equations are a foundational test-bed for algorithms related to numerical weather prediction and data assimilation (Asch et al., 2016; Reich and Cotter, 2015). They aim to represent a quantity of interest such as pressure along a slice of fixed latitude on the earth. We aim to test the reduced order models’ ability to perform both medium-range (five to 10 days) and long range ($>$ 10 days) forecasting (Kalnay, 2003).

While the Lorenz ’96 equations are widely used, it is extremely important to note that it has recently been shown that the equations do not represent a discretization of some partial differential equation model (van Kekem, 2018).

The Lorenz ’96 model has a known Kaplan-Yorke dimension of 27.1 and largest Lyapunov exponent of approximately $\lambda \approx 1.6852$, which were independently computed using a method found in (Dieci et al., 2011). The value of r is generally chosen based on computational limitations, and is thus not treated as a hyperparameter. For these reasons, we focus on the interesting case of $r = 28$ for the size of the reduced order model, though preliminary experiments were performed on other values of r.

We compare the SyCo-AE method eq. (11) proposed in this work with the standard autoencoder reduced order model eq. (6), with the linear DMD method eq. (8), and with the non-linear Quadratic manifolds eq. (9) method. Both the autoencoder-based methods are trained to minimize the full cost function eq. (19) (made up of eq. (18)) using the ADAM (Kingma and Ba, 2014) method with parameters $\beta_1 = 0.9$ and $\beta_2 = 0.95$, for 1000 epochs, with no minibatching, and a cyclic learning rate scheduler (Smith, 2017) in order to eliminate learning rate as a



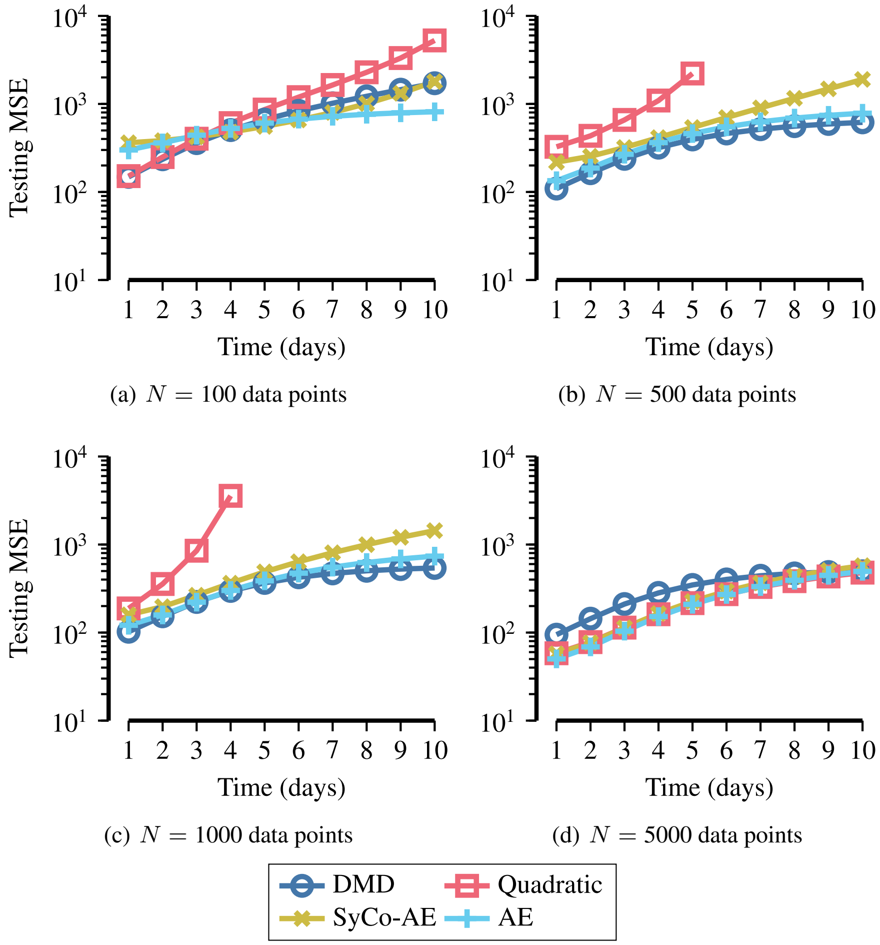
(a) $N = 100$ data points

(b) $N = 500$ data points

(c) $N = 1000$ data points

(d) $N = 5000$ data points

FIG. 3: Medium-range forecasting experiment for the Lorenz '96 equations and the corresponding ROMs of dimension $r = 28$ for various amounts of training data. For each subfigure, the x-axis represents the forecasting time in days, ranging from one to 10, and the y-axis represents the KL-divergence eq. (22) of the forecasted data points from the truth. Subfigure (a) has results for models trained on $N = 100$ data points, (b) on $N = 500$ data, (c) on $N = 1000$ data, and (d) on $N = 5000$ data.



(a) $N = 100$ data points

(b) $N = 500$ data points

(c) $N = 1000$ data points

(d) $N = 5000$ data points

FIG. 4: Medium-range forecasting experiment for the Lorenz ’96 equations and the corresponding ROMs of dimension $r = 28$ for various amounts of training data. For each subfigure, the x-axis represents the forecasting time in days, ranging from one to 10, and the y-axis represents the MSE eq. (21) of the forecasted data points from the truth. Subfigure (a) has results for models trained on $N = 100$ data points, (b) on $N = 500$ data, (c) on $N = 1000$ data, and (d) on $N = 5000$ data.

hyperparameter.

The data collected are trajectories of points collected six hours apart, with each trajectory starting 30 days from the start of the previous, with 0.2 time units corresponding to one day in the model. Not all data points and trajectories are utilized in order to test the methods in the small-data regime.

We train our models on various amounts of data points N. We take $N = 100$, $N = 500$, $N = 1000$ and $N = 5000$ data points, attempting to range from extremely small, to reasonably large. We take two different values of the roll out factor, K in eq. (18), namely $K = 1$ and $K = 9$. Note that N does not represent the amount of trajectories that we take, but instead represents the total amount of data points. For example, for $N = 100$, a roll out of $K = 1$ would correspond to 50 trajectories consisting of two data points each, while a roll out of $K = 9$ corresponds to 10 trajectories of 10 points each. In our testing the roll-out factor K did not play a significant role, thus all results are presented with the roll out factor $K = 1$, though data for $K = 9$ are available.

For both the autoencoder-based methods, we use a fully connected shallow neural network architecture as described in theorem 2, for the encoder θ, the decoder ϕ, and the reduced order dynamics f in eq. (6) and eq. (11). For the hidden layer size we take $H = 2000$, though models with $H = 500$ were also constructed with slightly worse performance. The encoder for the SyCo-AE method is additionally projected as in eq. (13). As theorem 2 requires a continuous activation function, we take the GELU function (Hendrycks and Gimpel, 2016) as our nonlinearity.

All computation was performed on a MacBook M2 Pro laptop, restricting the number of models that could be trained, thus not all hyperparameters could be optimized for.

For the first experiment we look at the qualitative features of all the methods, as compared to the true full order model for a long range forecast of 60 days. We take all models trained on $N = 5000$ data point with $K = 1$, and $H = 2000$ where applicable and forecast a single state outside of the training set.

The results, shown in fig. 2, reveal that the full order model exhibits quasi-periodic behavior, typical of chaotic systems. DMD eq. (8) exhibits decaying behavior, confirmed by the eigenvalues of $\mathbf{\Omega}$ all having negative real part. Quadratic manifold eq. (9) appears to exhibit the correct behavior for approximately 15 days, then appears to catastrophically diverge, which is a known problem in quadratic approximations to dynamical systems (Kaptanoglu et al., 2021). The standard autoencoder approach also appears to have the correct behavior for about five days, but also appears to catastrophically diverge. Only the SyCo-AE method appears to have the same quasi-periodic behavior as the full order model for the full 60 days of the long-range forecast.

In our final experiments with the Lorenz '96 equations we look at the quantitative difference in medium-range forecasting by the reduced order models. There is evidence to suggest that the Lorenz '96 system is ergodic (Fatkullin and Vanden-Eijnden, 2004), roughly meaning that the spatial mean is equivalent to temporal mean. To that end we make use of two different measures of error: the mean squared error (MSE) and the KL-divergence (Kullback and Leibler, 1951; Schulman, 2020). The MSE is defined by

$$\mathrm{MSE}(x) = \frac{1}{M}\sum_{i=1}^{M} ||x_i - x_i^{\text{true}}||_2^2, \tag{21}$$

with M representing the number of samples taken. The (approximate) KL-divergence of the data distribution p to the true distribution q is computed in the following numerically stable way,

$$\mathrm{D}_{\mathrm{KL}}(p \,\|\, q) \approx \frac{1}{M} \sum_{x_i \sim p, i=1}^{M} \frac{1}{2} (\log q(x) - \log p(x))^2, \tag{22}$$

with M again representing the number of samples taken, and the distributions approximated by Gaussian mixture models computed through kernel density estimation (Silverman, 1986).

We propagate $M = 10000$ states of the full order and reduced order models forward in time from one to 10 days, and calculate the MSE and KL-divergence over time for various data sizes. The results for the KL-divergence experiment in fig. 3 show a clear distinction between the four different methods. All methods perform poorly for $N = 100$ data points, with SyCo-AE eq. (11) being the most stable, though not by much. For $N = 500$ and $N = 1000$ data points, both DMD eq. (8) and SyCo-AE eq. (11) reach a steady state error, with SyCo-AE having a KL-divergence of an order of magnitude less. The standard autoencoder method is able to produce useful two day forecasts, but then diverges. For $N = 5000$ data points the quadratic manifolds method and the standard autoencoder are finally able to produce useful forecasts of about five to six days, but still catastrophically diverge afterwards.

The MSE results in fig. 4 do not reflect the same conclusions as the KL-divergence results. Almost all methods, except for the Quadratic ROM perform similarly in MSE. This is not a surprise as the training was meant to minimize the MSE.

The stark difference between the MSE and KL-divergence results indicates that merely looking at the MSE and ignoring other statistics can be misleading. The statistical forecasting utility of a given ROM method therefore cannot be judged solely by semblance of accuracy achieved for any one given data point, but must instead be judged by the accuracy of the predicted distributions.

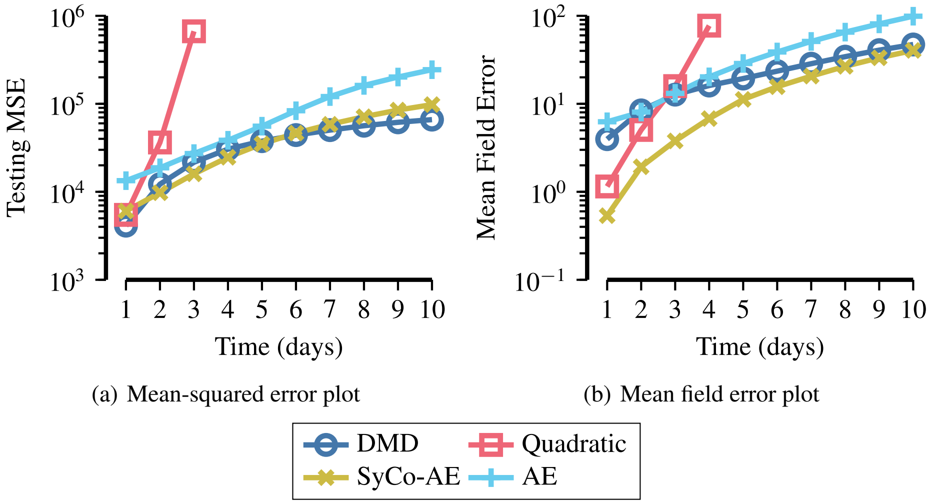
4.2 Quasi-Geostrophic Equations

For the next experiment, we look at a much larger chaotic system, the Quasi-geostrophic equations (Ferguson, 2008; Foster et al., 2013; Greatbatch and Nadiga, 2000; Majda and Wang, 2006). The equations model the flow of the streamfunction ψ, over the surface of the earth, and is described by the partical differential equation,

$$\begin{aligned} \omega_t + J(\psi, \omega) - Ro^{-1}\,\psi_x &= Re^{-1}\,\Delta\omega + Ro^{-1}\,F, \\ J(\psi, \omega) \equiv \psi_y\,\omega_x - \psi_x\,\omega_y, &\quad \omega = -\Delta\psi. \end{aligned} \tag{23}$$

For our parameters we take $Re = 450$ for the Reynolds number, $Ro = 0.0036$ for the Rossby number, and a spatial domain of $[0, 1] \times [0, 2]$ with homogeneous Dirichlet boundary conditions, which is discretized with 64 points in the x direction and 128 points in the y direction. and $F = \sin(\pi(y - 1))$ is the symmetric double gyre forcing term. The implementation of this problem was taken from the ODE Test Problems suite (Computational Science Laboratory, 2021; Roberts et al., 2021).

The experimental setup is similar to the Lorenz '96 equations in section 4.1. The same training parameters for the ADAM algorithm were taken. A shallow network was used for both the encoder and the decoder with a hidden layer dimension of $H = 500$. For the model specific parameters the reduced dimension size was taken to be $r = 25$, as this size was found to be stable for a model-informed quadratic ROM in (Popov and Sandu, 2022). The largest-Lyapunov exponent of



(a) Mean-squared error plot

(b) Mean field error plot

FIG. 5: Medium-range forecasting experiment for the Quasi-geostrophic equations and the corresponding ROMs of dimension $r = 25$. For each subfigure, the x-axis represents the forecasting time in days, ranging from one to 10. In subfigure (a) the y-axis represents the MSE eq. (21) of the forecasted data points from the truth, while in subfigure (b) the y-axis represents the mean field error eq. (25) of the forecasted data points from the truth.

the system was found to be $\lambda \approx 8.1097$ and is used in this work. We train the model on $N = 1000$ data points with $K = 1$ (thus 500 trajectories total). For this model one day is 0.0109 in model time.

As the QG equations are not ergodic, and the state space of the system is of dimension 8192 the KL-divergence testing used in section 4.1 would not produce meaningful results with current computational limitations. Instead of purely relying on the MSE eq. (21), we can look at the prediction of the mean field of the system,

$$\mathrm{MF}(x) = \frac{1}{M}\sum_{i=1}^{M} x_i, \tag{24}$$

and the corresponding mean field error,

$$\mathrm{MFE}(x) = \left\| \frac{1}{M}\sum_{i=1}^{M} x_i - \frac{1}{M}\sum_{i=1}^{M} x_i^{\text{true}} \right\|_2. \tag{25}$$

For this problem the mean field of the training data is propagated forward in time from one to 10 days for the full order and reduced order models, and the error between the mean field predictions is recorded.

The results over the testing data fig. 5 show a discrepancy between the best model in terms of MSE and the best model in terms of predicting the statistical mean field. The quadratic ROM is only good at predictions of one or two days, and falls apart beyond that. The AE ROM consistently performs worse than the linear DMD method. The SyCo-AE ROM performs better than DMD for two to five days of MSE predictions, but always outperforms DMD for predicting the mean field.

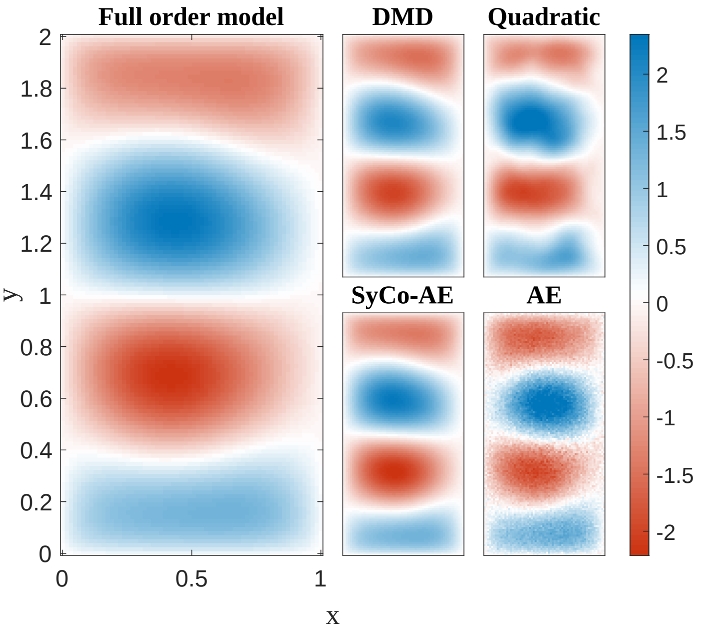


FIG. 6: Three day predicted mean field eq. (24) of the testing data for the Quasi-geostrophic equations for the full order model, DMD, the Quadratic ROM, SyCo-AE ROM, and the AE ROM.

A qualitative look at the three-day mean fields for the full order and reduced order models is presented in fig. 6. Both DMD and SyCo-AE ROM seem to behave in a similar manner closely matching the four major regions on the mean field. The AE ROM seems to not represent the spatial properties of the system well, having a ‘grainy’ image, meaning that the spatial derivatives of the system are not being preserved. The Quadratic ROM seems to have a growing instability in one of the regions of the mean field, and has severe visual discrepancies between itself and the true mean field.

5. CONCLUSIONS

We provided a new framework for reduced order modeling of chaotic systems with neural networks, synthetically constrained autoencoders (SyCo-AE) eq. (11) that augments a standard autoencoder-based ROM eq. (6) with a synthetic constraint. This simple synthetic constraint in the reduced space stands in as a proxy for an unknown constraint of the system in the full space. Thus, by learning the autoencoder and reduced order dynamics restricted to the simple manifold determined by the synthetic constraint, we implicitly learn the manifold of the full order system.

We test our method on the Lorenz ’96 equations and the Quasi-geostrophic equations. Results show that the SyCo-AE approach can create stable reduced order models capable of medium-range forecasting that are more accurate than methods that do not enforce a compact constraint on the total reduced order space. Additionally, results show that the amount of data needed to train useful reduced order models is significantly less than other non-linear methods, and seems to be the same as required to train a linear method.

Some of the limitations of this work are as follows. Alternative choices of the the synthetic constraint g are not explored, and a robust justification of the choice of the sphere is absent. A more robust exploration of the tunable hyperparameters and neural network architecture is required.

Future work would more formally explore the connection between a spherical synthetic constraint g and spherical embedding (Kosinski, 2013).

ACKNOWLEDGMENTS

The first author would like to thank Reid Gomillion and Steven Roberts for their insightful discussion on time integration methods. This work was sponsored in part by AFOSR (Air Force Office of Scientific Research) under contract number FA9550-22-1-0419.

REFERENCES

Aggarwal, C.C. , Neural Networks and Deep Learning, *Springer*, vol. **10**, no. 978, p. 3, 2018.

Asch, M., Bocquet, M., and Nodet, M., *Data Assimilation: Methods, Algorithms, and Applications*, SIAM, 2016.

Ascher, U.M. and Petzold, L.R., *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*, Vol. 61, Siam, 1998.

Baydin, A.G., Pearlmutter, B.A., Radul, A.A., and Siskind, J.M., Automatic Differentiation in Machine Learning: a Survey, *Journal of Marchine Learning Research*, vol. **18**, pp. 1–43, 2018.

Benner, P., Ohlberger, M., Cohen, A., and Willcox, K., *Model Reduction and Approximation: Theory and Algorithms*, SIAM, 2017.

Brunton, S.L. and Kutz, J.N., *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*, Cambridge University Press, 2022.

Brunton, S.L., Proctor, J.L., and Kutz, J.N., Discovering Governing Equations from Data by Sparse Identification of Nonlinear Dynamical Systems, *Proceedings of the national academy of sciences*, vol. **113**, no. 15, pp. 3932–3937, 2016.

Chada, N.K., Jasra, A., and Yu, F., Multilevel Ensemble Kalman-Bucy Filters, *arXiv preprint arXiv:2011.04342*, 2020.

Chen, R.T., Rubanova, Y., Bettencourt, J., and Duvenaud, D.K., Neural Ordinary Differential Equations, *Advances in neural information processing systems*, vol. **31**, 2018.

Computational Science Laboratory, ODE Test Problems, vol. **1**, no. 1, 2021. URL `https://github.com/ComputationalScienceLaboratory/ODE-Test-Problems`

Dieci, L., Jolly, M.S., and Van Vleck, E.S., Numerical Techniques for Approximating Lyapunov Exponents and Their Implementation, *Journal of Computational and Nonlinear Dynamics*, vol. **6**, no. 1, 2011.

Farmer, J.D., Ott, E., and Yorke, J.A., The Dimension of Chaotic Attractors, *Physica D: Nonlinear Phenomena*, vol. **7**, no. 1-3, pp. 153–180, 1983.

Fatkullin, I. and Vanden-Eijnden, E., A Computational Strategy for Multiscale Systems with Applications to Lorenz 96 Model, *Journal of Computational Physics*, vol. **200**, no. 2, pp. 605–638, 2004.

Ferguson, J., 2008. A Numerical Solution for the Barotropic Vorticity Equation Forced by an Equatorially Trapped Wave. Master's thesis, University of Victoria.

Foster, E.L., Iliescu, T., and Wang, Z., A Finite Element Discretization of the Streamfunction Formulation of the Stationary Quasi-Geostrophic Equations of the Ocean, *Comput. Methods Appl. Mech. Engrg.*, vol. **261**, pp. 105–117, 2013.

Geelen, R., Wright, S., and Willcox, K., Operator Inference for Non-Intrusive Model Reduction with Quadratic Manifolds, *Computer Methods in Applied Mechanics and Engineering*, vol. **403**, p. 115717, 2023.

Goodfellow, I., Bengio, Y., and Courville, A., *Deep Learning*, MIT press, 2016.

Greatbatch, R.J. and Nadiga, B.T., Four-Gyre Circulation in a Barotropic Model with Double-Gyre Wind Forcing, *J. Phys. Oceanogr.*, vol. **30**, no. 6, pp. 1461–1471, 2000.

Guckenheimer, J. and Holmes, P., *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*, Vol. 42, Springer Science & Business Media, 2013.

Hairer, E., Hochbruck, M., Iserles, A., and Lubich, C., Geometric Numerical Integration, *Oberwolfach Reports*, vol. **3**, no. 1, pp. 805–882, 2006.

Hairer, E., Nørsett, S.P., and Wanner, G., *Solving Ordinary Differential Equations. I, Nonstiff Problems*, Springer-Vlg, 1991.

Hairer, E. and Wanner, G., *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems*, Vol. 14, Springer-Vlg, 1996.

Heinonen, J. , *Lectures on Analysis on Metric Spaces*, Springer Science & Business Media, 2001.

Hendrycks, D. and Gimpel, K., Gaussian Error Linear Units (Gelus), *arXiv preprint arXiv:1606.08415*, 2016.

Hoel, H., Shaimerdenova, G., and Tempone, R., Multilevel Ensemble Kalman Filtering based on a Sample Average of Independent EnKF Estimators, *Foundations of Data Science*, p. 0, 2019.

Jaynes, E.T., *Probability Theory: The Logic of Science*, Cambridge university press, 2003.

Kalnay, E., *Atmospheric Modeling, Data Assimilation and Predictability*, Cambridge university press, 2003.

Kaptanoglu, A.A., Callaham, J.L., Aravkin, A., Hansen, C.J., and Brunton, S.L., Promoting Global Stability in Data-Driven Models of Quadratic Nonlinear Dynamics, *Physical Review Fluids*, vol. **6**, no. 9, p. 094401, 2021.

Karpatne, A., Kannan, R., and Kumar, V., *Knowledge Guided Machine Learning: Accelerating Discovery Using Scientific Knowledge and Data*, CRC Press, 2022.

Khalil, H.K., *Nonlinear Control*, Vol. 406, Pearson New York, 2015.

Kingma, D.P. and Ba, J., Adam: A Method for Stochastic Optimization, *arXiv preprint arXiv:1412.6980*, 2014.

Kitchin, R. and Lauriault, T.P., Small Data in the Era of Big Data, *GeoJournal*, vol. **80**, pp. 463–475, 2015.

Kosinski, A.A., *Differential Manifolds*, Courier Corporation, 2013.

Kullback, S. and Leibler, R.A., On Information and Sufficiency, *The annals of mathematical statistics*, vol. **22**, no. 1, pp. 79–86, 1951.

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

Lorenz, E.N., Predictability: A Problem Partly Solved, *Proc. Seminar on predictability*, Vol. 1, 1996.

Lou, A., Lim, D., Katsman, I., Huang, L., Jiang, Q., Lim, S.N., and De Sa, C.M., Neural Manifold Ordinary Differential Equations, *Advances in Neural Information Processing Systems*, vol. **33**, pp. 17548–17558, 2020.

Majda, A.J. and Wang, X., *Nonlinear Dynamics and Statistical Theories for Basic Geophysical Flows*, Cambridge University Press, Cambridge, 2006.

Parker, T.S. and Chua, L., *Practical Numerical Algorithms for Chaotic Systems*, Springer Science & Business Media, 2012.

Popov, A.A. and Sandu, A., 2022. Multifidelity data assimilation for physical systems. *Data Assimilation for Atmospheric, Oceanic and Hydrologic Applications (Vol. IV)*. Springer, pp. 43–67.

Popov, A.A. and Sandu, A., Multifidelity Ensemble Kalman Filtering using Surrogate Models Defined by Theory-Guided Autoencoders, *Data-driven modeling and optimization in fluid dynamics: From physics-based to machine learning approaches*, vol. **16648714**, p. 41, 2023.

Popov, A.A., Sarshar, A., Chennault, A., and Sandu, A., A Meta-Learning Formulation of the Autoencoder Problem, *arXiv preprint arXiv:2207.06676*, 2022.

Qin, T., Wu, K., and Xiu, D., Data Driven Governing Equations Approximation using Deep Neural Networks, *Journal of Computational Physics*, vol. **395**, pp. 620–635, 2019.

Rand, D., The Topological Classification of Lorenz Attractors, *Mathematical Proceedings of the Cambridge Philosophical Society*, Vol. 83, Cambridge University Press, pp. 451–460, 1978.

Reich, S. and Cotter, C., *Probabilistic Forecasting and Bayesian Data Assimilation*, Cambridge University Press, 2015.

Roberts, S., Popov, A.A., and Sandu, A., ODE Test Problems: a MATLAB Suite of Initial Value Problems, *arXiv preprint arXiv:1901.04098*, 2019.

Roberts, S., Popov, A.A., Sarshar, A., and Sandu, A., ODE Test Problems: a MATLAB Suite of Initial Value Problems, *arXiv preprint arXiv:1901.04098*, 2021.

Schulman, J., Approximating KL Divergence, 2020.
URL http://joschu.net/blog/kl-approx.html

Silverman, B.W., *Density Estimation for Statistics and Data Analysis*, Vol. 26, CRC press, 1986.

Smith, L.N., Cyclical Learning Rates for Training Neural Networks, *2017 IEEE winter conference on applications of computer vision (WACV)*, IEEE, pp. 464–472, 2017.

Stengel, R.F., *Optimal Control and Estimation*, Courier Corporation, 1994.

Strogatz, S.H., *Nonlinear Dynamics and Chaos with Student Solutions Manual: With Applications to Physics, Biology, Chemistry, and Engineering*, CRC press, 2018.

Uy, W.I.T., Hartmann, D., and Peherstorfer, B., Operator Inference with Roll Outs for Learning Reduced Models from Scarce and Low-Quality Data, *arXiv preprint arXiv:2212.01418*, 2022.

van Kekem, D.L., Dynamics of the Lorenz-96 model, PhD thesis, University of Groningen, 2018.

Weng, L., From Autoencoder to Beta-VAE, *lilianweng.github.io*, 2018.
URL https://lilianweng.github.io/posts/2018-08-12-vae/