

Learning Dissipative Chaotic Dynamics with Boundedness Guarantee

Sunbochen Tang^{a,1}, Themistoklis Sapsis^a, and Navid Azizan^{a,1}

This manuscript was compiled on October 27, 2025

Chaotic dynamics, commonly seen in weather systems and fluid turbulence, are characterized by their sensitivity to initial conditions, which makes accurate prediction challenging. Recent approaches have focused on developing data-driven models that preserve invariant statistics over long horizons since many chaotic systems observe dissipative behaviors and ergodicity. A crucial prerequisite for producing reliable statistics is that the model must generate bounded trajectories, a condition many models fail to guarantee, despite their empirical success. To address this fundamental challenge, we introduce a modular framework that enforces formal, provable guarantees of trajectory boundedness for neural network chaotic dynamics models. Our core contribution is a “dissipative projection” layer that leverages control-theoretic principles to ensure the learned system is dissipative. Specifically, our framework simultaneously learns a dynamics emulator and an energy-like function, where the latter is used to construct an algebraic dissipative constraint within the projection layer. A secondary benefit is that the learned invariant level set provides an outer estimate for the system’s strange attractor, which is known to be very difficult to characterize due to its complex geometry. We demonstrate our model’s ability to produce bounded long-horizon forecasts that preserve invariant statistics for chaotic dynamical systems including Lorenz 96 and a reduced-order model of the Kuramoto-Sivashinsky equation.

constrained machine learning | dynamical systems | turbulence

Chaos, characterized by exponential divergence after infinitesimal initial perturbations, is ubiquitous in a variety of complex dynamical systems, including climate models (1) and turbulence in fluids (2, 3). The exponential separation makes it challenging to accurately predict trajectories of chaotic systems. However, many chaotic systems of practical interest across various domains, including weather models and fluid dynamics (1, 2), turn out to be *dissipative*, meaning that their trajectories converge to a bounded and positively invariant set, often referred to as a strange attractor (4). Moreover, trajectories of dissipative chaos will visit almost every state on the attractor, resulting in ergodicity and invariant statistics (5). Consequently, rather than seeking pointwise-accurate predictions, the primary goal in modeling dissipative chaotic systems becomes capturing these invariant statistics over long forecast horizons.

Recent data-driven efforts have shown remarkable empirical success in building surrogate models that accelerate inference while preserving the long-term invariant statistics of dissipative chaos. These methods span a wide spectrum of structural assumptions and model complexity. On one end, structured nonlinear regression introduces physically motivated multi-level models to fit time series data efficiently (6, 7). At the other end, deep learning approaches rely on the representation power of neural networks to directly model complex chaotic behavior from raw data, while incorporating knowledge of shared physical system behaviors as specific architecture choices or regularization schemes (8–13). Hybrid approaches leverage autoencoder architectures to latent representation spaces where the dynamics evolve in simpler forms, with inspirations from Koopman theory (14), Dynamic mode decomposition (15), PCA (16), etc. Beyond one-step prediction, recurrent sequential models have also been explored to promote stability and improve forecast accuracy using more input information (17–19). In addition to standard recurrent models, a specific recurrent network architecture design for time series prediction, known as reservoir computing (RC), has demonstrated improved performance in reconstructing attractors in chaos and preserving invariant statistics (20–22).

To predict statistical properties on the attractor, data-driven models must generate arbitrarily long trajectories during inference to sufficiently sample the invariant measure. In practice, these models adopt an autoregressive paradigm that iteratively predicts the next state from its own prior outputs, making them

Significance Statement

Dissipative chaos commonly occurs in scientific domains from fluid dynamics to climate systems. Despite the empirical success of data-driven models in capturing their long-term statistics, these models often fail to satisfy a crucial requirement for producing reliable statistics, which is guaranteed trajectory boundedness. Here we address this issue by introducing a framework that provides formal guarantees of trajectory boundedness for neural network models. By simultaneously learning a dynamics model and an energy representation from data, we construct a dissipative projection layer that enforces energy dissipation as a provable, mathematical constraint rather than an empirical heuristic. This work demonstrates how embedding first principles can build a new class of trustworthy and physically consistent AI models for scientific discovery.

Author affiliations: ^aMassachusetts Institute of Technology, Cambridge, MA 02139

Please provide details of author contributions here.

Please declare any competing interests here.

¹To whom correspondence should be addressed. E-mail: {tangsun,azizan}@mit.edu

vulnerable to drifting outside the region of training data. Consequently, even though these models often demonstrate strong empirical performance in preserving invariant statistics, they are still prone to producing unbounded trajectories and invalid statistical forecasts. For more structured models such as multi-level quadratic regression models, theoretical analysis in (23) establishes pathological instability in their statistical solutions. A fundamental difficulty in using RNNs to model chaotic systems is revealed in (17), where it was mathematically proved that the gradients of RNNs diverge during the training process. Although it is generally difficult to theoretically analyze the behaviors of data-driven machine learning models, they are also found to generate diverging trajectories, resulting in invalid statistical solutions. Recent advanced time-series modeling approaches, such as RCs and Fourier Neural Operators, have been reported to experience the same fundamental issue in practice (20, 24, 25).

The core challenge in data-driven modeling for chaotic dynamical systems we are aiming to address is to mitigate unstable model behaviors in generating unbounded trajectories. Current proposed solutions generally fall into two categories: the first focuses on empirical stabilization, and the second seeks formal guarantees at the cost of generality. Within the first category, some techniques are architecture-specific, such as the use of noise-inspired regularization (26) or network pruning (27) to improve the long-term stability of reservoir computing models. Broader approaches aim to preserve the system's invariant statistics by constraining models to match key dynamical properties like Lyapunov exponents (28) or by adding statistics-based regularization to the loss function (29, 30). While these methods often reduce blowups in practice, they lack formal guarantees of boundedness. The second category of methods seeks to establish such guarantees by directly incorporating physical principles. However, these methods require restrictive model structures or substantial domain knowledge. For instance, energy conservation can be enforced structurally through the mathematical construction of quadratic regression models (7). A more recent approach for Fourier Neural Operators proposes a flexible strategy to encourage energy dissipation through a combination of regularization and post-training modifications (25). However, while broadly applicable, this method does not establish first-principles formal guarantees; instead, it relies on heuristics that require system-specific hyperparameter tuning and prior knowledge to be effective. Consequently, a critical gap remains for a method that can establish general-purpose boundedness guarantees without heavy reliance on system-specific prior knowledge or model-dependent tuning.

In this paper, we bridge this gap by introducing a framework that provides formal, first-principles guarantees of trajectory boundedness in neural network chaotic dynamics models. To the best of our knowledge, this is the first work to establish such guarantees for data-driven chaotic forecasting. Our core contribution is a novel "dissipative projection" layer that enforces dissipativity as a hard, mathematical constraint, consequently establishing trajectory boundedness guarantees. Rather than relying on system-specific domain knowledge, our approach learns both the underlying dynamics and the governing energy function directly from data. In particular, we leverage Lyapunov stability theory to derive efficient

algebraic conditions characterizing dissipativity, which are then embedded within the dissipative projection layer to guarantee stability during training and inference. As a secondary benefit, the learned energy function invariant set provides a tight outer approximation of the strange attractor, by regularizing the volume of such set. We illustrate this framework using the Lorenz 63 system, and then demonstrate its effectiveness in preserving invariant statistics and generating bounded trajectories for Lorenz 96 and a reduced-order model of the Kuramoto–Sivashinsky (KS) equation, highlighting its broad applicability across different chaotic benchmarks.

Data-driven Dissipative Chaos Modeling

Consider a chaotic dynamical system described by the finite-dimensional ODE,

$$\dot{x}(t) = f(x(t)), \quad [1]$$

where $x(t) \in \mathbb{R}^n$ represents the state of the dynamical system at time t , and $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is a nonlinear function that governs the dynamics. Many chaotic systems are *dissipative*, meaning that their trajectories eventually enter a bounded positively invariant set on which they exhibit chaotic behaviors. On this invariant set, also known as the strange attractor, the system becomes ergodic and consequently produces well-defined long-term statistical properties.

Definition 1. *We say that the system in Eq. 1 is dissipative if there exists a bounded and positively invariant set $M \subset \mathbb{R}^n$ such that*

$$\lim_{t \rightarrow \infty} \text{dist}(x(t), M) = 0, \quad \text{dist}(x(t), M) = \inf_{y \in M} \|x(t) - y\|$$

*In other words, every trajectory of the system will converge to M asymptotically, and stays within M once it enters. M is said to be **globally asymptotically stable**.**

The practical significance of studying dissipative chaotic systems is the possibility to shift the focus from the challenging task of predicting exact chaotic trajectories to the more tractable goal of capturing their invariant statistical properties over long-term trajectory forecasts. In data-driven modeling for dissipative chaos, our objective becomes constructing a neural network dynamics emulator, $\hat{f} : \mathbb{R}^n \rightarrow \mathbb{R}^n$, that generates prediction trajectories $\{\hat{x}(t)\}_{t=t_0}^{t_0+T}$ over a future time horizon T by solving the initial value problem,

$$\dot{\hat{x}}(t) = \hat{f}(\hat{x}(t)), \quad \hat{x}(t_0) = x(t_0), \quad [2]$$

from which the invariant statistics of the true system can be reproduced. The premise of reproducing meaningful statistical properties is that the trajectory remains bounded for a sufficiently long forecast horizon T . Although empirical success has been reported in a variety of recently proposed autoregressive models (8, 20), these models are prone to finite-time blowup in long trajectory rollouts, as illustrated in Fig. 1, which prohibits their use for statistical evaluation. This happens because the training dataset covers only a subset of the strange attractor, and without explicit guarantees, errors

*The definition is derived based on the one introduced by (4), where they stated dissipativity as every trajectory will enter M eventually. Here, we quantify this behavior using the notion of asymptotic stability and trajectory distance to M .

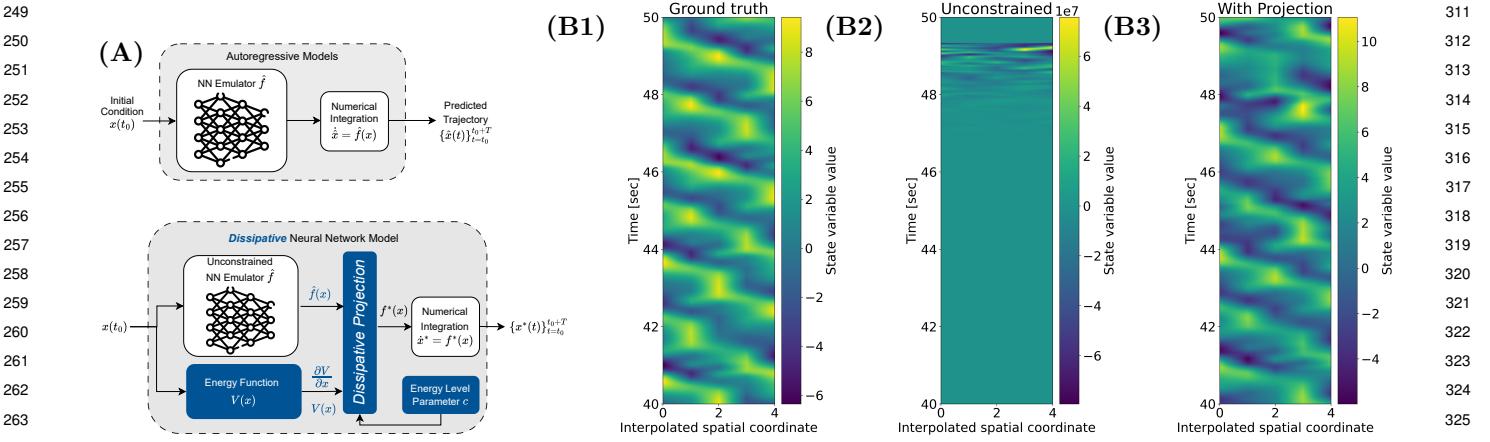


Fig. 1. Overview of our approach and main results. (A) Current autoregressive models without constraints suffer from accumulated error, leading to trajectories growing unbounded over extended rollout. Our approach overcomes this issue by building a dissipative projection layer that ensures the model is dissipative and guarantees bounded trajectories. (B1) Ground truth Hovmöller diagram of the Lorenz 96 system (with linear interpolation in the spatial dimension). (B2) An unconstrained network generates an unbounded trajectory during test time, whereas our model produces stable and statistically meaningful trajectories (B3).

accumulate when the model encounters states outside the training set, leading to divergence. Moreover, the exponential separation characteristic of chaotic systems embedded in the training data further amplifies these errors, contributing to divergence in model rollouts during test time.

To address this challenge, our approach introduces a *dissipative projection* layer to enforce an energy-based constraint into the neural network, which guarantees trajectories always remain bounded, ensuring their validity for long-term statistics evaluation. As illustrated in Fig. 1(A), our *dissipative NN model* jointly learns a dynamics emulator, an energy function, and an energy level parameter directly from data. This approach forces the learned dynamics model $\dot{x}^* = f^*(x)$ to be dissipative and also provides an outer estimate for the strange attractor as an energy level set $\{x : V(x) \leq c\}$. In numerical experiments for the Lorenz 96 system (31), while we observe that the unconstrained model in (8) generated unbounded trajectories during test time, our constrained model produces bounded trajectories that match the statistics patterns seen in the ground truth Hovmöller (32) diagram. This result exemplifies the importance of enforcing dissipativity for reliable long-term predictions in dissipative chaos modeling. Beyond ensuring trajectory boundedness, as illustrated in Fig. 2, since the true dynamics are dissipative, by only constructing models that are certified to be dissipative, this constraint effectively narrows the search space for neural network parameters. This is especially helpful in the limited data regime, where it becomes difficult for the model to learn the dissipative behaviors by simply fitting labeled data under trajectory prediction settings.

Dissipative Dynamics: A Control-theory Perspective

To develop models with inherent dissipativity, we first need to understand theoretical conditions that make a dynamical system dissipative. Note that dissipativity describes a system's energy behavior over time, independent of whether the system is chaotic or not. In this section, we focus on deriving algebraic conditions that are computationally efficient through the connection between dissipativity and

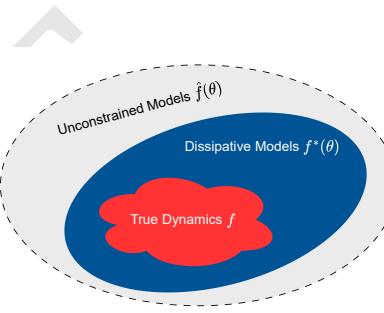


Fig. 2. The dissipative constraint effectively narrows the search space of model parameters because the true dynamics of interest are dissipative.

energy, which are crucial for our proposed architecture that guarantees dissipativity.

Energy-based Conditions for Dissipativity. Dissipativity, as the name suggests, has a close relationship with energy in a dynamical system. Intuitively, a dissipative system will lose energy over time, which corresponds to the trajectory converging to a bounded set. Although the trajectory convergence behavior is quantitatively stated in Definition 1, given a system $\dot{x} = f(x)$ known to be dissipative with access to the true dynamics f , it is still challenging to quantify the set M that the trajectory converges to. In the specific context of dissipative chaos, there has been a body of literature that tries to address this issue by studying invariant manifolds, volume contraction, and attempting to characterize the strange attractor (4). Despite the rigorous treatment and the progress over the years that help us understand the strange attractor, these characterizations are often stated in abstract mathematical concepts and a descriptive manner that is intractable to computationally verify, e.g., (33). Given our goal of enforcing dissipativity in neural network models, it is crucial to first derive computationally efficient conditions that ensure a system is dissipative.

In control theory, the concept of Lyapunov functions has been used extensively to formalize asymptotic stability of dynamical systems, which are also known as “energy-like”

functions due to strong connections with the mechanical energy of the system. More importantly, by leveraging the level set of such functions, numerous computationally tractable conditions have been derived and extensively used in designing practical controllers to ensure a system's asymptotic stability to equilibrium points (34). By generalizing asymptotic stability with respect to an equilibrium point to a level set of a Lyapunov function, we derive computationally efficient conditions that ensure dissipativity in a dynamical system.

Recall in Definition 1, a dissipative system requires the existence of a bounded set M , which satisfies (1) M is an invariant set (2) the system is globally asymptotically stable towards M . By reducing the definition to the existence of a Lyapunov function V and choosing M to be a level set of V , i.e., $M(c) = \{x : V(x) \leq c\}$ where $c > 0$ corresponds to the energy level, we derive the following conditions for invariance and asymptotic stability of $M(c)$ in Proposition 1 and 2, respectively.

Proposition 1 (invariant level set). *For a dynamical system in Eq. 1, suppose there is a continuously differentiable scalar-valued function $V : \mathbb{R}^n \rightarrow \mathbb{R}$ and a constant $c > 0$, such that*

$$\forall x \in \{x \in \mathbb{R}^n : V(x) > c\}, \dot{V}(x) = \frac{\partial V}{\partial x} f(x) \leq 0. \dagger$$

Then the level set $M(c) = \{x : V(x) \leq c\}$ is a positively invariant set for the system Eq. 1.

Proposition 2 (asymptotic stability). *For a dynamical system in Eq. 1, suppose there is a lower-bounded continuously differentiable scalar-valued function $V : \mathbb{R}^n \rightarrow \mathbb{R}$ and a constant $c > 0$, such that*

- (1) $\forall x \in \{x \in \mathbb{R}^n : V(x) > c\}, \dot{V}(x) < 0$;
- (2) V is radially unbounded.

Then the level set $M(c) = \{x : V(x) \leq c\}$ is globally asymptotically stable.

The proofs for both propositions are included in the SI Appendix. We include the following illustrative figures to sketch the intuitions behind the conditions in Proposition 1 and 2.

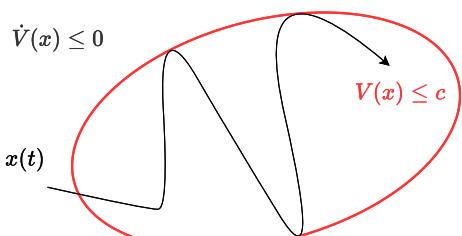


Fig. 3. The level set boundary serves as a barrier since the trajectory cannot gain energy outside. Once entering $M(c)$, the trajectory will be confined within.

[†]Here $\frac{\partial V}{\partial x}$ refers to the row vector $[\frac{\partial V}{\partial x_1}, \dots, \frac{\partial V}{\partial x_n}]$.

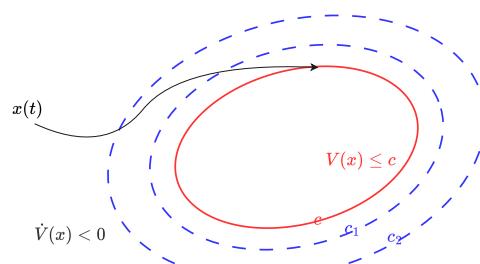


Fig. 4. The trajectory loses energy over time outside $M(c)$ because $\dot{V}(x) \leq 0$, resulting in convergence to the level set $M(c)$ eventually ($c_2 > c_1 > c > 0$).

Algebraic Conditions for Dissipativity and Attractor Outer Estimation.

Compared to Definition 1, the conditions derived in the propositions above are much more quantitative. However, verifying these conditions for a given system in Eq. 1 computationally is not trivial; for example, the negative semi-definite condition for $\dot{V}(x)$ is defined only on a certain part of the state space, outside the level set $M(c)$ in Proposition 1. Inspired by the S-procedure in sum-of-squares programming (35), we can replace these conditions with a single algebraic condition that is independent of the state, which is more computationally tractable.

Theorem 1. *Suppose there exists a lower-bounded radially unbounded C^1 function $V : \mathbb{R}^n \rightarrow \mathbb{R}$ and a constant $c > 0$ such that for the dynamical system in Eq. 1,*

$$\forall x \in \mathbb{R}^n, \dot{V}(x) + V(x) - c \leq 0. \quad [3]$$

Then the system Eq. 1 is dissipative and $M(c)$ is globally asymptotically stable.

Proof. The condition Eq. 3 implies that $\forall x \in \mathbb{R}^n$ such that $V(x) > c$, $\dot{V}(x) \leq -(V(x) - c) < 0$. Therefore, by Proposition 1 and 2, the level set $M(c)$ is both globally asymptotically stable and positively invariant. \square

Strange Attractor Outer Estimation. Recall the characteristics of the strange attractor discussed in the background section; if only the first two properties are satisfied by a set S , then this set is known as “the attractor” (36). The difference from the strange attractor is that the attractor does not require the exponential separation of neighboring trajectories everywhere, hence it is also a superset of the strange attractor. Note that if a set $S \subset M(c)$ is an attractor, then $M(c)$ is both invariant and globally asymptotically stable. Therefore, in addition to certifying a system to be dissipative, the level set $M(c)$ also provides an outer approximation of the original attractor, which is consequently an outer estimate of the strange attractor as well.

In practice, characterizing the strange attractor is challenging due to its complex geometry (33). In addition to empirically reproducing the strange attractor using the learned model similar to (25, 29), our method also provides a level set outer estimate for the strange attractor by learning the Lyapunov function V and level set parameter c .

497 **Importance of the Algebraic Condition.** The
 498 condition in Eq. 3 is strictly stronger than the conditions in
 499 Proposition 1 and 2. Although the condition is slightly more
 500 conservative, the computational tractability of obtaining a
 501 state-independent algebraic condition is well worth the trade-
 502 off. More importantly, this algebraic condition is crucial for
 503 constructing our proposed model architecture that ensures
 504 the learned system is dissipative, which will be discussed in
 505 detail in the next section.

508 Inherently Dissipative Neural Network Dynamics Model

509 Now we propose our architecture that simultaneously learns
 510 a dynamics emulator and a Lyapunov function V , with the
 511 former guaranteed to be dissipative by enforcing the condition
 512 in Eq. 3 through the construction of a projection layer. The
 513 overall structure is illustrated in Fig. 1(A).

514 Our proposed model consists of three learnable com-
 515 ponents: a neural network dynamics emulator \hat{f} that
 516 approximates the true nonlinear dynamics f on the right-
 517 hand side of Eq. 1; a quadratic energy-like function $V(x) =$
 518 $(x - x_0)^T Q(x - x_0)$ serving as the Lyapunov function with
 519 learnable parameters Q (Q is parameterized to be positive
 520 definite) and x_0 that define the shape and center of the level
 521 sets, respectively; and a level set parameter $c > 0$. Using
 522 these three components and the gradient information $\partial V / \partial x$,
 523 we construct a “dissipative projection” layer that outputs a
 524 predicted dynamics emulator $f^*(x) \in \mathbb{R}^n$ which guarantees
 525 dissipativity of its corresponding dynamics $\dot{x} = f^*(x)$.

526 A key feature of our method is its modularity, which
 527 provides the flexibility to enforce stability as an “addon”
 528 scheme. The proposed “dissipative projection” layer is
 529 agnostic to the architectural choice for the dynamics
 530 emulator, which allows the integration of any suitable function
 531 approximator as the backbone model \hat{f} . Similarly, while
 532 we use a quadratic form for the Lyapunov function V for
 533 simplicity, our framework accommodates any function that
 534 satisfies the conditions outlined in Theorem 1. This flexibility
 535 establishes our approach as a general-purpose pipeline for
 536 enforcing stability, rather than a solution dependent on a
 537 specific class of models.

538
 539
 540 **Dissipative Projection Ensures Boundedness.** Following The-
 541 orem 1, if $\dot{x} = f^*(x)$ satisfies the condition in Eq. 3, then the
 542 system is guaranteed to be dissipative and converge to the
 543 level set $M(c)$ defined by the learned Lyapunov function $V(x)$
 544 and the learned constant $c > 0$. The dissipative projection
 545 layer is designed to modify the backbone emulator \hat{f} such that
 546 the output f^* produces a learned dynamical model $\dot{x} = f^*(x)$
 547 that satisfies the condition in Eq. 3, therefore certified to be
 548 dissipative.

549 Intuitively, this condition informs a subspace for the vector
 550 field $f(x)$ in which the forward dynamics will be dissipative.
 551 The dissipative projection layer is designed to project any
 552 unconstrained dynamics approximator, $\hat{f}(x)$, into such a
 553 subspace to ensure dissipativity.

554 More specifically, given an input $x \in \mathbb{R}^n$, the dissipative
 555 projection layer output $f^*(x)$ is chosen as the vector closest
 556 to the approximator $\hat{f}(x)$ under ℓ^2 distance in the subspace of
 557 \mathbb{R}^n defined by Eq. 3, i.e., $f^*(x)$ is the solution to the following

558 optimization problem:

$$f^*(x) = \operatorname{argmin}_{f(x)} \|f(x) - \hat{f}(x)\|_2^2 \quad [4a]$$

$$\text{subject to } \frac{\partial V}{\partial x} f(x) + V(x) - c \leq 0 \quad [4b]$$

559 As discussed extensively earlier, the constraint of the
 560 optimization problem, which is adopted from the condition
 561 in Eq. 3, ensures the learned dynamics emulator $f^*(x)$ to be
 562 dissipative while being computationally efficient. In addition
 563 to the fact that the condition is easily verifiable through
 564 basic arithmetic operations, the constraint is also linear in
 565 the optimization variable $f(x)$. Since the above optimization
 566 problem has a quadratic loss and a linear constraint, an
 567 explicit solution can be found and computed using ReLU
 568 activation, similar to the approach in (37, 38):

$$f^*(x) = \hat{f}(x) - \frac{\partial V^T \operatorname{ReLU}\left(\frac{\partial V}{\partial x} \hat{f}(x) + V(x) - c\right)}{\|\frac{\partial V}{\partial x}\|^2} \quad [5]$$

569 With the dissipative projection layer implemented as in
 570 Eq. 5, we state the following corollary that formalizes the
 571 dissipativity of our proposed model architecture, which is a
 572 direct result of Theorem 1. A computational proof verifying
 573 $f^*(x)$ satisfies Eq. 3 $\forall x \in \mathbb{R}^n$ is included as well.

574 **Corollary 1.** *The learned dynamics model $\dot{x} = f^*(x)$ is a
 575 dissipative system with a bounded and positively invariant
 576 level set $M(c) = \{x : V(x) \leq c\}$. The set $M(c)$ is globally
 577 asymptotically stable, which implies every trajectory of the
 578 system is bounded and converges to $M(c)$ asymptotically.*

579 *Proof.* If $\frac{\partial V}{\partial x} \hat{f}(x) + V(x) - c \leq 0$, it follows from Eq. 5 that
 580 $f^*(x) = \hat{f}(x)$. Therefore, in this case, $\frac{\partial V}{\partial x} f^*(x) + V(x) - c \leq 0$,
 581 meaning that $f^*(x)$ satisfies the condition in Eq. 3.

582 If $\frac{\partial V}{\partial x} \hat{f}(x) + V(x) - c > 0$, then we have the following,

$$\begin{aligned} & \frac{\partial V}{\partial x} f^*(x) + V(x) - c \\ &= \frac{\partial V}{\partial x} \left[\hat{f}(x) - \frac{\partial V^T \operatorname{ReLU}\left(\frac{\partial V}{\partial x} \hat{f}(x) + V(x) - c\right)}{\|\frac{\partial V}{\partial x}\|^2} \right] + V(x) - c \\ &= \frac{\partial V}{\partial x} \hat{f}(x) - \left(\frac{\partial V}{\partial x} \hat{f}(x) + V(x) - c \right) + V(x) - c = 0 \end{aligned}$$

583 which ensures that $f^*(x)$ satisfies the stability condition Eq. 3.
 584 Therefore, by Theorem 1, the system $\dot{x} = f^*(x)$ is dissipative
 585 and $M(c)$ is globally asymptotic stable.

586 Additionally, since $\forall x \notin M(c)$, $\dot{V}(x) < -(V(x) - c) <$
 587 0, the trajectory always loses energy outside $M(c)$. Also
 588 note that if the trajectory starts within $M(c)$, it can never
 589 leave, i.e., $V(x(t)) \leq c$ for all $t \geq 0$. Therefore, for any $t \in$
 590 $[0, \infty)$, $V(x(t)) = V(x(0)) + \int_0^t \dot{V}(x(\tau)) d\tau \leq \max\{V(x(0)), c\}$.
 591 Since every level set of $V(x(t))$ is bounded (see proof for
 592 Proposition 2 in SI Appendix), $x(t)$ is always bounded. \square

593
 594 **Training with Invariant Set Volume Regularization.** We con-
 595 sider a training dataset consisting of trajectory points which
 596 are evenly sampled at h [sec] from a few ground truth
 597 trajectories initialized at randomly sampled initial conditions.
 598 Unlike (25, 29), we do not assume the trajectories in the
 599 training set are already inside the attractor, which allows
 600 for more flexibility when learning unknown chaotic systems,
 601 where the transition period before it reaches an invariant

statistics is unknown. This is beneficial for our proposed model to learn where to place the invariant set $M(c)$ and apply dissipative projection.

During training, we consider a multi-step setting, where we roll out the learned model for T steps, each step sampled at h [sec] using a numerical integration scheme. More specifically, given an initial condition chosen from the training dataset $x_0^{(i)}$, we forward simulate the learned system $\dot{\hat{x}}^{(i)} = f^*(\hat{x}^{(i)})$ with $\hat{x}^{(i)}(0) = x_0^{(i)}$ and obtain sampled states at the same sampling period h [sec], $\hat{x}_k^{(i)} = \hat{x}^{(i)}(kh)$ at $k = 1, 2, \dots, T$. By sampling N such trajectory snapshots of length T from the training dataset, we define the prediction loss as the MSE between the predicted rollout sequence $(\hat{x}_k^{(i)})_{k=1}^T$ and the ground truth sequence $(x_k^{(i)})_{k=1}^T$: Prediction Loss = $\frac{1}{NT} \sum_{i=1}^N \sum_{k=1}^T \|x_k^{(i)} - \hat{x}_k^{(i)}\|_2^2$.

In the dissipative projection layer, the quadratic Lyapunov function $V(x)$ and the level set parameter c both need to be optimized during training. Although the prediction loss depends on V and c through the projection operator that produces $f^*(x)$, optimizing only the prediction loss may not be a well-defined optimization problem. More specifically, if we have found a level set $M(c_1)$ that is globally asymptotically stable and invariant, then any superset $M(c_2)$ for $c_2 > c_1 > 0$ is also globally asymptotically stable and invariant. Therefore, there could potentially be infinitely many solutions that lead to the small prediction loss.

To address this issue, we introduce a regularization loss that encourages the learned level set $M(c)$ to be as small as possible, which aligns with our goal of characterizing a tight outer estimate of the strange attractor. Toward this objective, we use the volume of the ellipsoid $M(c)$ as the regularization loss. Combining the prediction and regularization loss, we have the following training loss with a weight hyperparameter $\lambda > 0$ for balancing the regularization terms:

$$\text{Loss} = \frac{1}{NT} \sum_{i=1}^N \sum_{k=1}^T \|x_k^{(i)} - \hat{x}_k^{(i)}\|_2^2 + \lambda \text{Vol}(M(c)), \quad [6a]$$

$$\text{Vol}(M(c)) = \frac{\pi^{n/2}}{\Gamma(\frac{n}{2} + 1)} \sqrt{\frac{c^n}{\det(Q)}} \quad [6b]$$

Results

Unlike most data-driven methods that prioritize data over structures, our approach aligns the ML model with physics principles by learning both the dynamics and an energy representation and enforcing an energy dissipation constraint. We illustrate our method and demonstrate the effectiveness of providing formal trajectory boundedness guarantees in reliably reproducing invariant statistics through a set of numerical experiments based on Lorenz 63, Lorenz 96, and Kuramoto–Sivashinsky (KS) equation, with an increasing level of complexity. For more quantitative evaluations regarding invariant statistics and more visualizations, please refer to the SI Appendix.

Learned flow is dissipative towards the invariant energy level set. We first consider the classic Lorenz 63 system, originally proposed in (1) as a simplified model for atmospheric convection. The low-dimensionality of the system makes it easy to visualize both the strange attractor and dissipative

behaviors of the system flow. As shown in Fig. 5a, a 50,000-step long-term trajectory rollout generated by our model (“fstar”) accurately reproduces the characteristic “butterfly-shaped” strange attractor of the ground truth (“GT”). The learned invariant energy level set, visualized as a yellow ellipsoid with a red dot indicating the learned center x_0 , provides a tight outer-estimate for the strange attractor, and the trajectory quickly enters the level set and stays in afterwards.

Furthermore, Fig. 5b illustrates the learned model’s dissipative behavior by comparing its projected flow map on the $x_1 - x_2$ plane with the ground truth. The learned flow not only matches the ground truth near the attractor but also consistently points inward from outside the learned invariant set, confirming the model’s dissipativity by design. Overall, the Lorenz 63 example demonstrates our method’s ability to learn geometrically interpretable and stable models, which guarantee to generate bounded trajectories reconstructing invariant statistics.

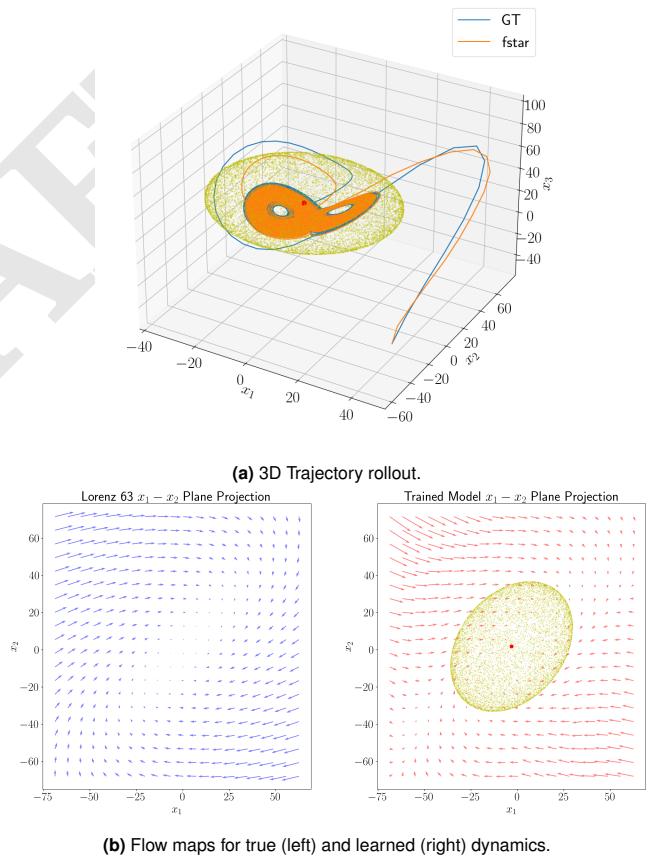


Fig. 5. Lorenz 63 (a) Trajectories generated by the learned model (“fstar”) and true dynamics (“GT”) are visualized along with the learned invariant set (sampled with yellow points), where the red dot is the center of the set. (b) Comparison of the flow map projected onto $x_1 - x_2$ plane.

Boundedness guarantee prevents finite-time blowup in high-dimensional systems. To demonstrate the necessity of guaranteed boundedness in data-driven methods, we consider two higher-dimensional chaotic systems: the 5-dimensional Lorenz 96 system and a 32-dimensional reduced-order model of the KS equation (KS-ROM). For these more complex

systems, unconstrained machine learning models experience finite-time blow-ups, resulting in prediction trajectories invalid for statistics evaluations.

As illustrated in Fig. 1(A), we benchmark against an unconstrained machine learning baseline which uses a standard multilayer perceptron (MLP) architecture to approximate the dynamics f by \hat{f} . In the finite-dimensional case, the MLP baseline is equivalent to the Neural Operator (39) and the DeepONet (11), both are state-of-the-art data-driven methods for general PDE modeling that do not incorporate any stability or energy constraints. For fair comparisons, we use the same MLP backbone for our method, augmenting it with an energy representation and the dissipative projection. For both Lorenz 96 and the reduced-order model for KS (KS-ROM), we train the unconstrained MLP and our proposed model on the exact same datasets, such that any difference in performance would be a direct consequence of the stability constraints. During testing, we sample a random initial condition, and forward simulate both the unconstrained model, $\dot{x} = \hat{f}(x)$, and our model with dissipative projection $\dot{x} = f^*(x)$ in an autoregressive manner (illustrated in Fig. 1(A)), and compare the trajectory rollouts generated by these two models. Since these system dimensions exceed three, we use Principal Component Analysis (PCA) to project the trajectory rollouts onto their first two principal components, such that a straightforward 2D visualization of the higher-dimensional trajectories can be presented.

The testing trajectories generated by the unconstrained model \hat{f} completely deviate from the attractor and then exhibit finite-time blow-up for both Lorenz 96 and KS-ROM systems, as illustrated in Fig 6(A1, B1) respectively, by scatter plots comparing the first two PCA components of the generated trajectories with their corresponding ground truth trajectories. In contrast, our proposed model with dissipative projection guarantees trajectory boundedness, producing prediction trajectories that recover the shape of the strange attractor for both systems, as shown in Fig 6(A2) and (B2). Moreover, by applying the same PCA to sampled points on the boundary of the invariant level set $V(x) \leq c$, Fig. 6(A2, B2) not only verify the trajectory quickly converges to our learned level set (illustrated as the red point cloud), but also show that the level set form a tight outer-estimate of the strange attractor.

Enforcing boundedness helps preserve invariant statistics. A key goal in modeling dissipative chaos is to reproduce the invariant statistics of the system's strange attractor, a task that requires generating bounded trajectories. When training data is limited, unconstrained models often fail by producing divergent trajectories, rendering statistical evaluation meaningless. Our method overcomes this by ensuring trajectory boundedness without prior knowledge of the system's statistics, unlike other approaches that incorporate known measures into the loss function (29, 30). We achieve this by constraining trajectories to a learned level set that acts as an outer estimate of the attractor, guiding the dynamics to the relevant region of the state space. To demonstrate the efficacy of this constraint, we compare rollouts from our model against an unconstrained baseline using: (1) spatiotemporal plots to illustrate flow patterns and (2) histograms of the leading principal components to show improved statistical distributions. Further visualizations,

including Fourier spectra, Fourier modes, and learned energy characteristics, are provided in the SI Appendix.

The spatiotemporal plots in Fig. 1(B) and Fig. 7 validate that our model generates trajectories that recover system characteristics while the unconstrained baseline fails and diverges unboundedly. For the Lorenz 96 system, the Hovmöller plot (32) shows that the unconstrained model's solution diverges to unbounded values. Our model, however, maintains a stable rollout that correctly reproduces the westward-propagating wave patterns seen in the ground truth. Similarly, for the reduced-order model of the Kuramoto-Sivashinsky equation, the solution reconstructed from 32 Fourier modes shows a finite-time blowup for the unconstrained model. In contrast, our model's trajectory remains bounded and successfully captures the complex emergence of cellular chaos and coarsening events characteristic of the true dynamics.

To quantitatively evaluate the preservation of invariant statistics, we project the long-horizon testing trajectory rollouts onto their first two principal components (PCs) and compute the probability density function (PDF) of the projected points. These leading PCs capture dominant modes of variance and provide a low-dimensional visualization of the geometric structure of the attractor. The 2D histograms in Fig. 6(A3, B3) compare the PDFs of the ground truth (left), unconstrained model (\hat{f} , middle), and our proposed model (f^* , right). In the region near the attractor, the unconstrained model's trajectory distribution is scattered and unstructured, failing to capture the distinct characteristics of the ground truth attractor. In contrast, our model not only confines the trajectory to the correct region of the subspace but also reproduces the intricate shape of the invariant measure.

We provide a quantitative comparison of the statistical property prediction errors between the unconstrained model \hat{f} and our constrained model f^* for both Lorenz 96 and KS-ROM systems. First, the Kullback-Leibler (KL) divergence between the predicted and the true PCA distributions further validates the visual assessment in Fig. 6(A3, B3): our model's divergence (Lorenz 96: 0.4078; KS-ROM: 0.8355) is significantly lower than that of the unconstrained model (Lorenz 96: 9.7856, KS-ROM: 3.0557) which fails to capture the attractor's geometric characteristics. Second, we evaluated the relative error of the Fourier energy spectra acquired from the predicted trajectory rollout compared to ground truth. Here again, the unconstrained model exhibits orders of magnitude higher error due to finite-time blowup (Lorenz 96: 164.10; KS-ROM: 244.47). In stark contrast, our proposed model's spectrum matches the ground truth closely (Lorenz 96: 0.0213; KS-ROM: 0.0001). These metrics demonstrate that our method effectively preserves the invariant statistics of chaotic systems by ensuring trajectory boundedness through dissipative constraints, even when trained on limited data and without prior knowledge of the system's statistical properties.

Conclusion

In this paper, we propose a novel neural network architecture for learning chaotic systems that ensures the learned dynamics are dissipative, which guarantees the model to always generate bounded trajectories and provide meaningful statistics evaluation. By leveraging control theoretic ideas, we have derived algebraic conditions that ensure dissipativity and embed these conditions into the neural network through

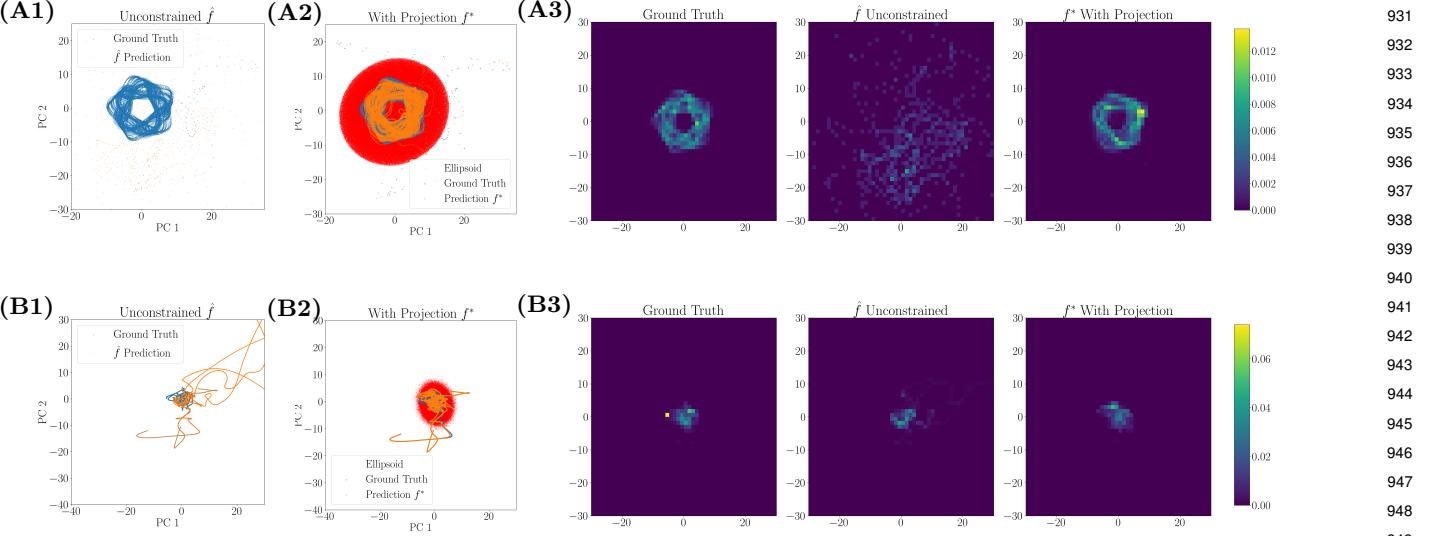


Fig. 6. Principal component analysis (PCA) comparison between trajectories generated by unconstrained model (\hat{f}) and our proposed model with dissipative projection (f^*) for Lorenz 96 (Row A) and a reduced-order KS model (Row B). (A1, B1) Unconstrained model generates a trajectory that quickly deviates from the attractor and then grows unbounded. (A2, B2) Our proposed model provides boundedness guarantees, which guides the generated trajectory to enter and traverse the strange attractor. In addition, the learned invariant set (red ellipsoid point cloud) provides a tight outer-estimate of the attractor. (A3, B3) The 2D histograms represent the probability density of trajectories in the PCA leading components for Lorenz 96 and the reduced-order KS model. In both cases, the unconstrained model \hat{f} (middle) produces an unstructured distribution that scatters across the PCA space, while our proposed model f^* (right) is able to reproduce the shape and density of the ground truth distribution (left), which validates its capability to better preserve the system's invariant statistics.

a projection layer design. Using the level set of an energy function learned simultaneously with the dynamics emulator, our model also provides an outer estimate for the strange attractor, which is difficult to characterize due to its complex geometry. Numerical experiments for chaotic systems, including Lorenz 96 and reduced-order KS model, show the model's capability of preserving invariant statistics of the true dynamics. Experiments also show that a model trained without the dissipative projection layer leads to finite-time blowup and unreliable statistics, which exemplifies the importance of our proposed approach in learning dissipative chaotic dynamics.

Materials and Methods

Numerical Models of Chaotic Systems. We used three chaotic systems of increasing complexity for our numerical experiments. Ground truth data for all systems was generated by integrating the governing equations using the 4th-order Runge-Kutta (RK4) method with a time step of $h = 0.01$ seconds.

Lorenz 63 System. The Lorenz 63 system (1) is a 3-dimensional ordinary differential equation (ODE) given by:

$$\begin{aligned}\dot{x}_1 &= \sigma(x_2 - x_1), \\ \dot{x}_2 &= x_1(\rho - x_3) - x_2, \\ \dot{x}_3 &= x_1x_2 - \beta x_3,\end{aligned}$$

where the state is $x \in \mathbb{R}^3$. We used the standard chaotic parameters $\sigma = 10.0$, $\beta = 8/3$, and $\rho = 28.0$.

Lorenz 96 System. The Lorenz 96 system (31) is a higher-dimensional ODE. We used a 5-dimensional version described as follows, with the forcing constant set to $F = 8.0$ to generate chaotic behavior.

$$\dot{x}_i = (x_{i+1} - x_{i-2})x_{i-1} - x_i + F, \quad i = 1, \dots, 5,$$

where $x_{-1} = x_4$, $x_0 = x_5$, $x_6 = x_1$.

Kuramoto-Sivashinsky (KS) Reduced-Order Model (ROM). We used a 32-dimensional ODE model derived from the Kuramoto-Sivashinsky (KS) partial differential equation (2). The reduction was performed using the Galerkin projection method (40, 41). Full details on the derivation are provided in the SI Appendix.

Model Architecture and Training.

Model Architecture. Our proposed model, f^* , and the unconstrained baseline model, \hat{f} , share an identical multilayer perceptron (MLP) backbone to ensure a fair comparison.

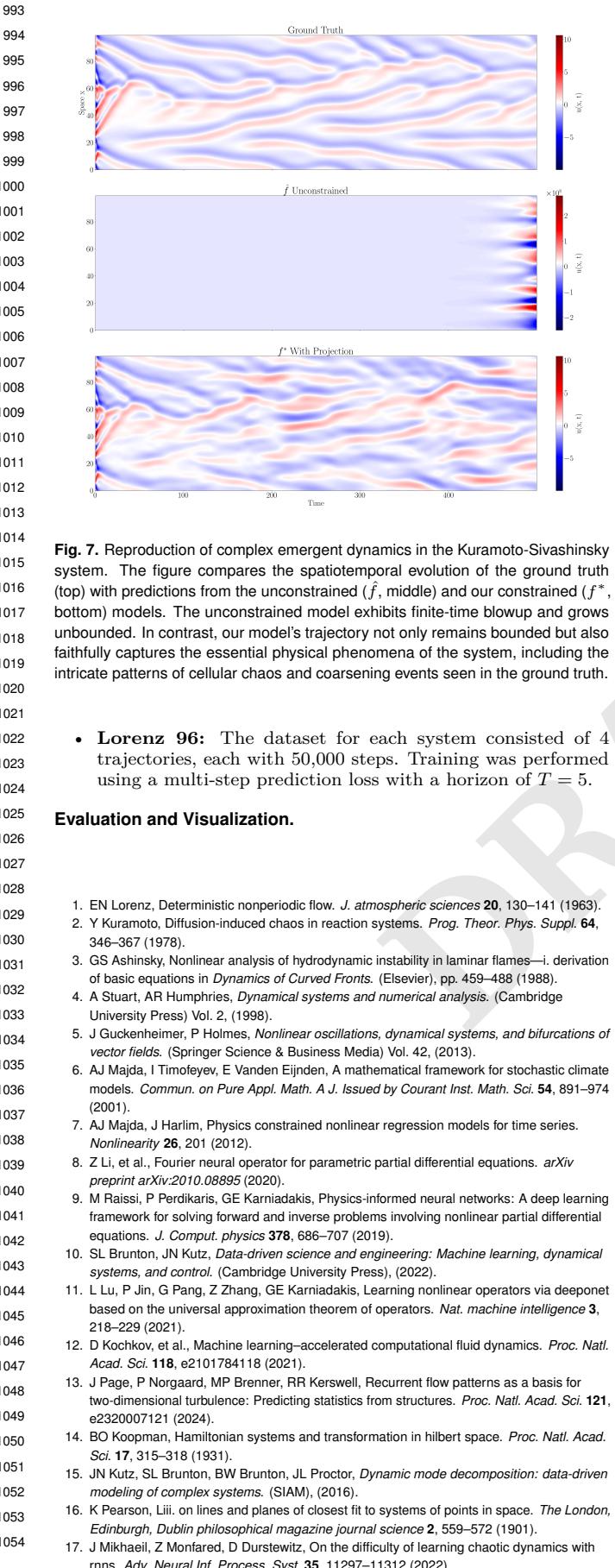
The MLP backbone \hat{f} consists of two hidden layers and uses a GelU (42) activation function for all three systems. For the Lorenz 63 and 96 systems, each hidden layer contains 64 neurons. For the reduced-order KS (KS-ROM) system, the hidden layer width increases to 512 neurons as the dynamics the model tries to capture become much more complex.

Our model, f^* , augments this backbone with an additional network to represent the energy function $V(x)$ and a final projection layer that enforces the energy dissipation constraint. The energy function $V(x)$ is parameterized as a quadratic function $V(x) = (x - x_0)^T Q(x - x_0)$, where the positive-definite matrix Q and center x_0 are learned. Note that Q is constructed to be positive definite, as the learnable parameters are log-diagonal terms and off-diagonal terms of a lower triangular matrix L forming the Cholesky decomposition $Q = LL^T$.

Training Datasets and Procedure. All models were trained using the Adam optimizer with a learning rate of 1×10^{-4} for 30,000 epochs. For our proposed model, the level set parameter c is initialized at a sufficiently large number, and the regularization weight parameter λ can be chosen over a hyperparameter grid search over $10^{-1}, 10^{-2}, \dots, 10^{-8}$.

The baseline MLP and our constrained model were trained on the exact same datasets for each system:

- **Lorenz 63:** The dataset consisted of 10 trajectories, each with 200 steps. Training was performed using a multi-step prediction loss with a horizon of $T = 5$.
- **Lorenz 96:** The dataset for each system consisted of 4 trajectories, each with 500 steps. Training was performed using a single-step prediction loss.



- **Lorenz 96:** The dataset for each system consisted of 4 trajectories, each with 50,000 steps. Training was performed using a multi-step prediction loss with a horizon of $T = 5$.

Evaluation and Visualization.

Trajectory Rollouts and Simulation. For testing, long-horizon trajectories were generated by simulating the learned models autoregressively from a random initial condition not seen during training. Trajectory integration was performed using the RK4 method with a time step of $h = 0.01$ seconds. For the Lorenz 63 and the KS-ROM system, 10 trajectories of 50,000 steps were generated for creating visualizations and evaluating the statistical properties of the learned systems. For Lorenz 96, 20 trajectories of 20,000 steps were generated for evaluation.

Statistical Metrics. We used two quantitative metrics to evaluate the models' ability to reproduce the invariant statistics of the true system from the long-horizon rollouts.

- **KL Divergence:** The Kullback-Leibler (KL) divergence was calculated between the 2D probability density function (PDF) of the model's trajectory projected onto its first two principal components and the PDF of the ground truth trajectory. The PDFs were estimated using 2D histograms with 50×50 bins.
- **Spectrum Error:** The Fourier energy spectrum was first computed for the long-horizon trajectory rollouts. To quantify the discrepancy, we calculated the relative L2 norm error between the predicted spectrum (S_{pred}) and the ground truth spectrum (S_{true}). This error is given by the formula:

$$\text{Error} = \frac{\|S_{\text{pred}} - S_{\text{true}}\|_2}{\|S_{\text{true}}\|_2}$$

where $\|\cdot\|_2$ denotes the standard Euclidean norm. This metric measures the overall deviation of the predicted spectrum's shape and magnitude relative to that of the ground truth.

ACKNOWLEDGMENTS. The authors acknowledge the MIT SuperCloud and Lincoln Laboratory Supercomputing Center for

providing computing resources that have contributed to the results reported within this paper. This work was supported in part by MathWorks, the MIT-IBM Watson AI Lab, the MIT-Azure Science Hub, and the MIT-Google Program for Computing Innovation.

- EN Lorenz, Deterministic nonperiodic flow. *J. atmospheric sciences* **20**, 130–141 (1963).
- Y Kuramoto, Diffusion-induced chaos in reaction systems. *Prog. Theor. Phys. Suppl.* **64**, 346–367 (1978).
- GS Ashinsky, Nonlinear analysis of hydrodynamic instability in laminar flames—i. derivation of basic equations in *Dynamics of Curved Fronts*. (Elsevier), pp. 459–488 (1988).
- A Stuart, AR Humphries, *Dynamical systems and numerical analysis*. (Cambridge University Press) Vol. 2, (1998).
- J Guckenheimer, P Holmes, *Nonlinear oscillations, dynamical systems, and bifurcations of vector fields*. (Springer Science & Business Media) Vol. 42, (2013).
- AJ Majda, I Timofeyev, E Vanden Eijnden, A mathematical framework for stochastic climate models. *Commun. on Pure Appl. Math. A J. Issued by Courant Inst. Math. Sci.* **54**, 891–974 (2001).
- AJ Majda, J Harlim, Physics constrained nonlinear regression models for time series. *Nonlinearity* **26**, 201 (2012).
- Z Li, et al., Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895* (2020).
- M Raissi, P Perdikaris, GE Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. physics* **378**, 686–707 (2019).
- SL Brunton, JN Kutz, *Data-driven science and engineering: Machine learning, dynamical systems, and control*. (Cambridge University Press), (2022).
- L Lu, P Jin, G Pang, Z Zhang, GE Karniadakis, Learning nonlinear operators via deepenet based on the universal approximation theorem of operators. *Nat. machine intelligence* **3**, 218–229 (2021).
- D Kochkov, et al., Machine learning–accelerated computational fluid dynamics. *Proc. Natl. Acad. Sci.* **118**, e2101784118 (2021).
- J Page, P Norgaard, MP Brenner, RR Kerswell, Recurrent flow patterns as a basis for two-dimensional turbulence: Predicting statistics from structures. *Proc. Natl. Acad. Sci.* **121**, e2320007121 (2024).
- BO Koopman, Hamiltonian systems and transformation in hilbert space. *Proc. Natl. Acad. Sci.* **17**, 315–318 (1931).
- JN Kutz, SL Brunton, BW Brunton, JL Proctor, *Dynamic mode decomposition: data-driven modeling of complex systems*. (SIAM), (2016).
- K Pearson, Lii, on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, Dublin philosophical magazine journal science* **2**, 559–572 (1901).
- J Mikhaeil, Z Monfared, D Durstewitz, On the difficulty of learning chaotic dynamics with nnns. *Adv. Neural Inf. Process. Syst.* **35**, 11297–11312 (2022).
- PR Vlachas, W Byeon, ZY Wan, TP Sapsis, P Koumoutsakos, Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks. *Proc. Royal Soc. A: Math. Phys. Eng. Sci.* **474**, 20170844 (2018).
- M Sangiorgio, F Dercle, Robustness of lstm neural networks for multi-step forecasting of chaotic time series. *Chaos, Solitons & Fractals* **139**, 110045 (2020).
- Z Lu, BR Hunt, E Ott, Attractor reconstruction by machine learning. *Chaos: An Interdiscip. J. Nonlinear Sci.* **28** (2018).
- PR Vlachas, et al., Backpropagation algorithms and reservoir computing in recurrent neural networks for the forecasting of complex spatiotemporal dynamics. *Neural Networks* **126**, 191–217 (2020).
- E Boltt, On explaining the surprising success of reservoir computing forecaster of chaos? the universal machine learning dynamical system with contrast to var and dmd. *Chaos: An Interdiscip. J. Nonlinear Sci.* **31** (2021).
- AJ Majda, Y Yuan, Fundamental limitations of ad hoc linear and quadratic multi-level regression models for physical systems. *Discret. & Continuous Dyn. Syst. B* **17** (2012).
- J Pathak, Z Lu, BR Hunt, M Girvan, E Ott, Using machine learning to replicate chaotic attractors and calculate lyapunov exponents from data. *Chaos: An Interdiscip. J. Nonlinear Sci.* **27** (2017).
- Z Li, et al., Learning chaotic dynamics in dissipative systems. *Adv. Neural Inf. Process. Syst.* **35**, 16768–16781 (2022).
- A Wikner, BR Hunt, J Harvey, M Girvan, E Ott, Stabilizing machine learning prediction of dynamics: Noise and noise-inspired regularization. *arXiv preprint arXiv:2211.05262* (2022).
- A Haluszczynski, J Aumeier, J Hertoux, C Räth, Reducing network size and improving prediction stability of reservoir computing. *Chaos: An Interdiscip. J. Nonlinear Sci.* **30** (2020).
- JA Platt, SG Penny, TA Smith, TC Chen, HD Abarbanel, Constraining chaos: Enforcing dynamical invariants in the training of reservoir computers. *Chaos: An Interdiscip. J. Nonlinear Sci.* **33** (2023).
- R Jiang, PY Lu, E Orlova, R Willett, Training neural operators to preserve invariant measures of chaotic attractors. *Adv. Neural Inf. Process. Syst.* **36** (2024).
- Y Schiff, et al., Dyslim: Dynamics stable learning by invariant measure for chaotic systems. *arXiv preprint arXiv:2402.04467* (2024).
- EN Lorenz, Predictability: A problem partly solved in *Proc. Seminar on predictability*. (Reading), Vol. 1, pp. 1–18 (1996).
- E Hovmöller, The trough-and-ridge diagram. *Tellus* **1**, 62–66 (1949).

1117	33. J Milnor, On the concept of attractor. <i>Commun. Math. Phys.</i> 99 , 177–195 (1985).	1179
1118	34. H Khalil, <i>Nonlinear Systems</i> , Pearson Education. (Prentice Hall), (2002).	1180
1119	35. G Blekherman, PA Parrilo, RR Thomas, <i>Semidefinite optimization and convex algebraic geometry</i> . (SIAM), (2012).	1181
1120	36. SH Strogatz, <i>Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering</i> . (CRC press), (2018).	1182
1121	37. Y Min, A Sonar, N Azizan, Hard-constrained neural networks with universal approximation guarantees. <i>arXiv preprint arXiv:2410.10807</i> (2024).	1183
1122	38. Y Min, SM Richards, N Azizan, Data-driven control with inherent lyapunov stability in 2023 <i>62nd IEEE Conference on Decision and Control (CDC)</i> . (IEEE), pp. 6032–6037 (2023).	1184
1123	39. N Kovachki, et al., Neural operator: Learning maps between function spaces with applications to pdes. <i>J. Mach. Learn. Res.</i> 24 , 1–97 (2023).	1185
1124	40. YS Smyrlis, DT Papageorgiou, Computational study of chaotic and ordered solutions of the kuramoto-sivashinsky equation, Technical report (1996).	1186
1125	41. P Holmes, <i>Turbulence, coherent structures, dynamical systems and symmetry</i> . (Cambridge university press), (2012).	1187
1126	42. D Hendrycks, K Gimpel, Gaussian error linear units (gelus). <i>arXiv preprint arXiv:1606.08415</i> (2016).	1188
1127		1189
1128		1190
1129		1191
1130		1192
1131		1193
1132		1194
1133		1195
1134		1196
1135		1197
1136		1198
1137		1199
1138		1200
1139		1201
1140		1202
1141		1203
1142		1204
1143		1205
1144		1206
1145		1207
1146		1208
1147		1209
1148		1210
1149		1211
1150		1212
1151		1213
1152		1214
1153		1215
1154		1216
1155		1217
1156		1218
1157		1219
1158		1220
1159		1221
1160		1222
1161		1223
1162		1224
1163		1225
1164		1226
1165		1227
1166		1228
1167		1229
1168		1230
1169		1231
1170		1232
1171		1233
1172		1234
1173		1235
1174		1236
1175		1237
1176		1238
1177		1239
1178		1240