Chaos
An Interdisciplinary Journal
of Nonlinear Science

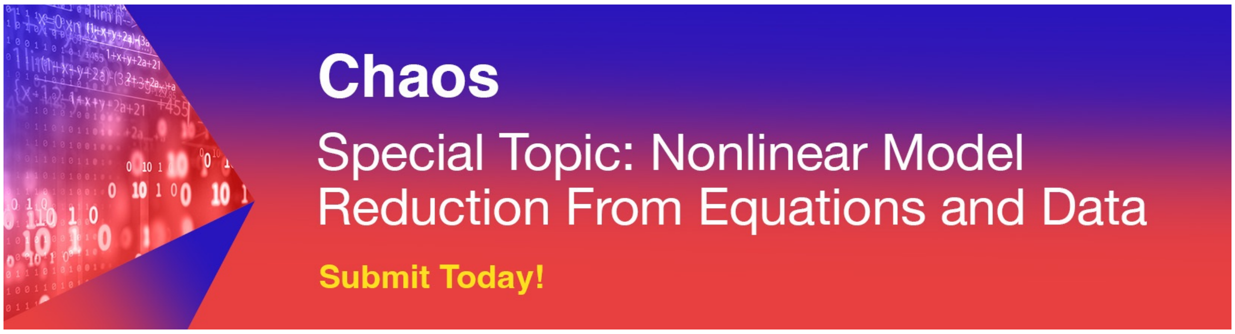
Data-driven modeling of nonlinear traveling waves

Cite as: Chaos **31**, 043128 (2021); https://doi.org/10.1063/5.0043255
Submitted: 07 January 2021 • Accepted: 05 April 2021 • Published Online: 19 April 2021

J. Koch

COLLECTIONS

EP This paper was selected as an Editor's Pick



AIP Publishing

Data-driven modeling of nonlinear traveling waves

Cite as: Chaos **31**, 043128 (2021); doi: 10.1063/5.0043255
Submitted: 7 January 2021 · Accepted: 5 April 2021 ·
Published Online: 21 April 2021

J. Koch[a)]

AFFILIATIONS

Oden Institute for Computational Engineering and Sciences, University of Texas at Austin, Austin, Texas 78712, USA

[a)] **Author to whom correspondence should be addressed:** james.koch@austin.utexas.edu

ABSTRACT

Presented is a data-driven machine learning framework for modeling traveling wave spatiotemporal dynamics. The presented framework is based on the steadily propagating traveling wave ansatz, $u(x,t) = U(\xi = x - ct + a)$. For known evolution equations, this coordinate transformation reduces governing partial differential equations to a set of coupled ordinary differential equations (ODEs) in the traveling wave coordinate ξ. Although traveling waves are readily observed in many physical systems, the underlying governing equations may be unknown. For these instances, the traveling wave dynamical system can be modeled empirically with neural ODEs. Presented are these ideas applied to several physical systems that admit traveling waves. Examples include traveling wave fronts, pulses, and wavetrains restricted to one-way wave propagation in a single spatial dimension. Last, applicability to real-world physical systems is presented with an exploration of data-driven modeling of rotating detonation waves.

Much in the same way one can seek traveling wave solutions to partial differential equations (PDEs), proposed is a modeling framework where one seeks a dynamical system in the traveling wave coordinate ξ that approximates an observed traveling wave profile. Thus, in this paper, the goal is to identify, model, and analyze the dynamics of underlying traveling wave ordinary differential equations (ODEs) directly from observations. For the specific task of modeling observed traveling waves, this methodology eliminates the need to identify, model, or otherwise analyze the governing PDE.

I. INTRODUCTION

The traveling wave is a fundamental structure that arises in many physical systems governed by partial differential equations (PDEs). Such structures are readily observed in fluid mechanics,[1] condensed-matter physics,[2] optics,[3] neuroscience,[4] and biology,[5,6] among many other fields.[7,8] The physics governing such systems are often complex, multi-component, multi-scale, and nonlinear; analysis of traveling waves is often limited to direct numerical simulation of fundamental physical laws as described by PDEs. However, for the steadily propagating one-dimensional (1D) wave—i.e., traveling with a constant velocity and shape—the *traveling wave ansatz* ($\xi = x - ct + a$) has long been used as an effective coordinate transformation that recasts the spatiotemporal dynamics of the governing PDE as a dynamical system in the single traveling wave coordinate ξ. The resulting dynamical system is often more mathematically tractable and readily analyzed using standard ordinary differential equation (ODE) techniques.

In this work, proposed is methodology that aims to approximate traveling wave dynamics in the traveling wave coordinate ξ—the latent space for such systems. This domain-specific knowledge is leveraged to define the inverse problem of extracting traveling wave models from data. The proposed methods are constructed around the central assumptions that (i) there exist traveling wave solutions to the governing physics and (ii) there exists a steady velocity for which such traveling waves are viable. In this article, presented are the applications of these ideas to representative systems that admit traveling wave solutions. In addition to reproducing the waveform, the resulting model systems are evaluated on their ability to reproduce the phase space in the neighborhood of the waveform trajectories. Where applicable, this includes the evaluation of the location, type, and stability of fixed points and location of system nullclines. Last, shown is application of these ideas to experimental observations of a rotating detonation engine whose steady-state operation features a collection of highly nonlinear traveling combustion waves.

II. BACKGROUND

The motivation for this study stems from the ongoing investigation of the physics of the *Rotating Detonation Engine* (RDE) whose hallmark feature during operation is a number of traveling combustion waves that consume combustible fuel mixture and expel exhaust at high velocity, producing useful thrust or otherwise high enthalpy flow. A full description of the device and its operation is given in Sec. IV D. Curiously, and seemingly contradictory, although the physics of compressible, shock-laden, and reactive fluid mechanics are complex, these devices prominently feature self-organization and mode-locking of combustion waves that persist in time. Despite achieving steady-state behavior (in the wave-attached reference frame) in many situations, to directly model these physics requires simulations with millions or billions of degrees of freedom that are integrated with time steps short enough to resolve the fastest physics and to a solution time long enough to fully develop the slowest physics. The result is a computational cost that is prohibitively large to many in the field. The need for data-driven *Reduced-Order Models* (ROMs) is clear; their adoption may play a critical role in speeding up the physics exploration, design, and analysis of these systems.

For physical systems that feature traveling waves, like the RDE, current data-driven ROM architectures and system identification techniques are insufficient. In this section, a brief review of ROM architectures is presented within the context of systems with underlying symmetries. A similar review of data-driven system discovery and surrogate modeling follows.

A. ROMs and symmetries

The proliferation of data has allowed for the recent rapid development of techniques and algorithms aimed at data-driven system identification and modeling.[9] In many physics problems, the spatiotemporal evolution of quantities of interest can be expressed by one or more coupled PDEs, which, if known, can rarely be solved analytically. A topical example is fluid dynamics as governed by the Navier–Stokes equations, where nonlinearities and a multitude of varying scales interact to form not only the complex flow fields associated with turbulence, but also coherent structures such as vortices, traveling waves, and shocks.[1] A typical workflow for modeling such systems is discretizing the PDE over the spatial domain of interest to create a high-dimensional coupled ODE that can be integrated in time. While such methods are robust and have been used with great success, they are often computationally expensive. This is especially true in fluid dynamic simulations where the interactions of scales contribute to the overall evolution of a flowfield; thus, all scales need to be properly resolved to provide an accurate solution.

Projection-based model order reduction techniques [those based on the *Singular Value Decomposition* (SVD)] seek to leverage underlying structure and patterns in the solution space of the high-dimensional full order model to perform future state predictions at a reduced computational cost. For oscillatory flows, techniques such as the *Proper Orthogonal Decomposition* (utilizing the SVD) have been successfully employed to extract the dominant space-time correlated structures.[10] Furthermore, these structures, or modes, can be used as a reduced basis set into which the governing PDE can be projected to create a ROM.[9] POD-Galerkin is one such ROM framework that utilizes a POD reduced basis.[11]

Standard projection-based model order reduction techniques for transport-dominated physics often suffer from a non-negligible slow modal energy decay associated with the POD.[9,12] This results in a large number of modes required to reproduce the physics. The failure of such methods is rooted in the inability of the SVD to handle translational and other symmetries (such as rotations) of the spatiotemporal field. Embedded within the POD is the assumption of separation of variables, where space and time can be cleanly decoupled. For many circumstances, this assumption may approximately hold for a reasonable time horizon. A traveling wave cannot be separated into time and spatial components; to construct a projection-based ROM without removing the translational symmetry results in a artificially high-order model.

Removing the translational symmetry, or "freezing" the traveling wave, is a straightforward approach to preconditioning input data such that projection-based methods can be applied with greater success. Methods such as the *Shifted Proper Orthogonal Decomposition* (sPOD[12]) and *Unsupervised Traveling Wave Identification with Shifting and Truncation* (UnTWIST[13]) aim to remove the translational symmetry by offsetting the drift of traveling waves such that they appear "frozen" in a new reference frame. For a steadily propagating wave, by removing the translational symmetry, the first POD mode corresponds to the waveform that is invariant in time. Thus, the waveform becomes the reduced basis upon which a ROM can be built. While these methods are extremely effective in future state predictions, they do not generalize well or offer physical insights into a given problem. Indeed, for a steadily propagating wave, any future state can be obtained by shifting the initial condition by the speed of the wave multiplied with the elapsed time. This is the linear representation of the wave, even though the physics may be inherently nonlinear. Furthermore, the applicability of these methods is dependent on the scope of the parameter space in which the training data were generated. For non-constant wave speeds and wave interactions, methods such as UnTWIST provide a robust means of disambiguating between waves and extracting their trajectories, thereby providing an unsupervised method of "freezing" the waves. Projection methods can then be applied in the non-constant reference frame. For wave interactions and instabilities, this may yield interpretable results regarding the waves' nonlinear dynamics.

B. Data-driven system discovery and surrogate modeling

Data proliferation can also allow for the "reverse engineering" of governing physics in the form of ordinary or partial differential equations. These tasks usually aim to create either (i) parsimonious and interpretable evolution equations or (ii) approximations to the state operator(s), a surrogate model, which can then be integrated in time with standard techniques. Methods exist for both model discovery and surrogate modeling of dynamical systems described by ODEs as well as PDEs.

Sparse Identification of Nonlinear Dynamics (SINDy[14]) performs a least-squares fit of data to a dynamical system constructed from a library of candidate functions from which a sparsity-promoting algorithm chooses a parsimonious representation of the dynamics. This is a particularly attractive method because the resulting model is easily interpreted and evaluated. Should the model indeed capture the physics correctly, it is by construction generalizable and can be parameterized. However, the quality of the resultant model is strongly dependent on the quantity, quality (presence of noise, especially after differentiation), and coordinate representation of the input data.

Neural ODEs (NODEs[15,16]) are a relatively new machine learning technique for approximating the right hand side of a differential equation with a neural network. NODE, while not as interpretable as a successful implementation of SINDy, still is a valuable method that can learn nonlinear interactions that are embedded within the input data without any *a priori* knowledge of the underlying system. These differ from standard time-series forecasting techniques (deep neural networks, residual networks, long short-term memory networks, etc.) in that the model is continuous in depth and that the neural network is integrated in time with standard ODE solvers; during training, residuals are backpropagated through both the network and the integrator. Thus, the surrogate model is the inseparable unit of the ODE solver and the neural network.

SINDy and NODEs are only two of many methods for dynamical system identification and modeling for systems described by ODEs. For PDEs, extensions of these methods have been developed and used with varying degrees of success. PDE-FIND[17] exploits the same sparsity-promotion ideas used in SINDy, though extended to include differential operators. Similarly, NODEs have been used in conjunction with convolutional neural networks to model separately the temporal and spatial evolution of a quantity of interest.[16] Leveraging pattern recognition techniques from the broader ML community, *Physics-Informed Neural Networks* (PINNs), have been used extensively to perform data-driven discovery of governing equations.[18] *Hidden Physics Models*[19] aim to learn governing equations from small data sets using Gaussian processes. These methods excel at identifying governing equations, though some may be more appropriate than others depending on quality, quantity, and orientation of available data.

C. Traveling wave dynamical systems

The one-dimensional (1D) *Traveling Wave Ansatz* (TWA) can be stated as seeking traveling wave solutions to a governing PDE via the transformation into the traveling wave coordinate $\xi = x - ct + a$, where c is the speed of an assumed right-running wave and a is a spatial offset. Through this transformation, a spatially (x) and temporally (t) varying quantity of interest u is recast as $u(x, t) = U(\xi = x - ct + a)$. Should the governing physics be known in the form of a PDE, the TWA is a powerful tool for recasting the governing equations as a coupled set of ODEs. By the chain rule, substitution into a governing PDE and differentiation is straightforward,

$$U_t = \frac{dU}{d\xi}\frac{d\xi}{dt} = -cU', \tag{1}$$

$$U_{tt} = \cdots = c^2 U'', \tag{2}$$

where the prime denotes differentiation with respect to ξ. Similarly, for spatial derivatives,

$$U_x = \frac{dU}{d\xi}\frac{d\xi}{dx} = U', \tag{3}$$

$$U_{xx} = \cdots = U''. \tag{4}$$

Upon substitution into a 1D wave equation, the resulting coupled system contains only ordinary derivatives with respect to the traveling wave coordinate ξ [as in Eq. (5)] and is thus amenable to standard ODE analysis techniques.

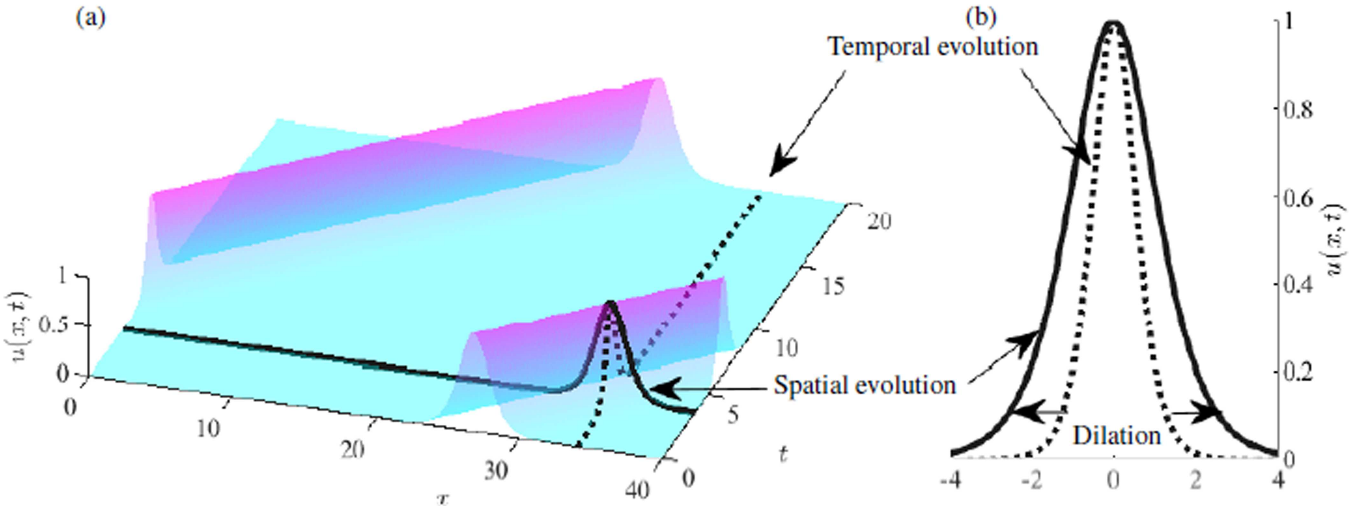


FIG. 1. A pictorial representation of the traveling wave anzatz. In (a), a KdV sech-pulse soliton ($c = 2$) travels on a periodic boundary. In (b), temporal and spatial slices are shown such that their peaks are centered at $t = 0$ and $x = 0$. For steadily propagating waves, the temporal and spatial evolution of $u(x, t)$ differ only by dilation of the temporal axis by the wave speed. Thus, for the profiles shown in (b), dilating the temporal evolution from t to ct (where $c = 2$ for this simulation) exactly recovers the spatial waveform.

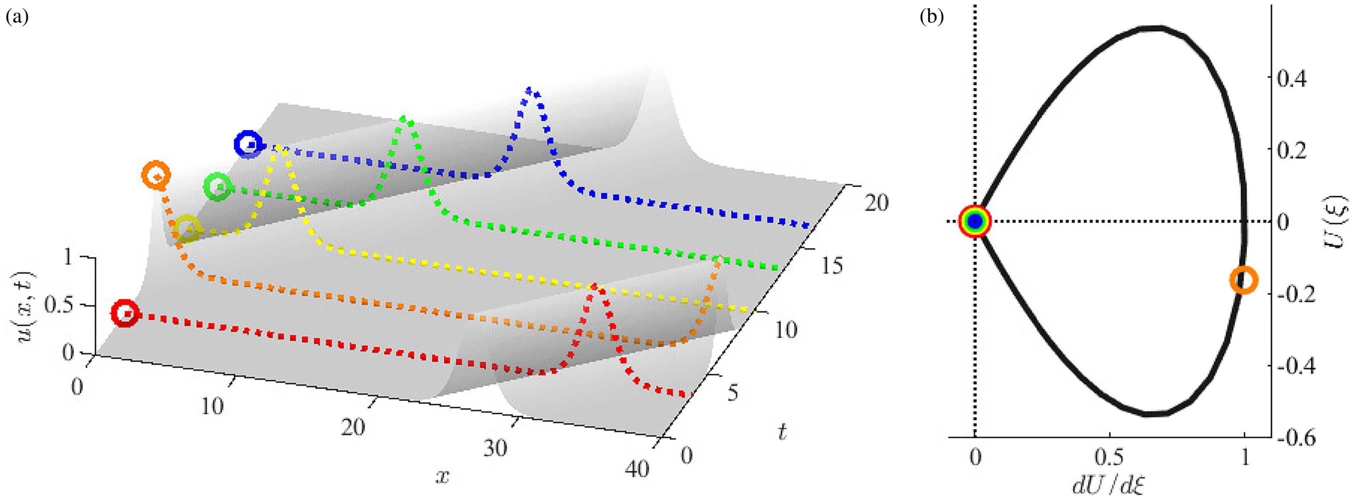


FIG. 2. For steadily propagating waves, snapshots of the spatiotemporal dynamics (a) map to a common trajectory through the phase space (b). This distance along the common trajectory in the phase space is measured by the coordinate $\xi = x - ct + a$. However, by re-defining $t_i = \tau = 0$ for each ith snapshot and choosing $a = 0$, the coordinate transform becomes $\xi_i = x - c\tau + a = x$. The colored circles in (a) show how values of $u(x, t)$ map to the phase plane of the system in (b).

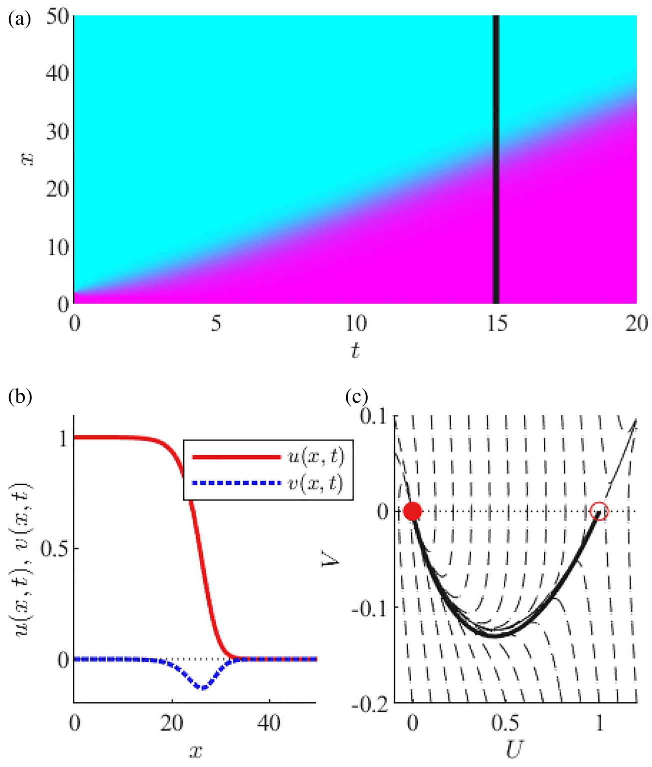


FIG. 3. (a) A KPP–Fisher wave front steadily propagates through the closed domain $x \in [0, 50]$. The steady waveform at time $t = 15$ [the vertical cut in (a)] is profiled in (b). This single snapshot is used as the training data for the surrogate neural ODE. The front is a heteroclinic orbit in the phase space, as shown in (c).

Central to the reduction of wave equations to ODEs is the relationship between spatial and temporal evolution for steadily propagating waves. As seen in Fig. 1, the evolution in space and time only differs by a proportionality constant—the speed of the wave. For example, for the Korteweg–de Vries soliton moving with speed $c = 2$ in Fig. 1, dilating the temporal axis by the speed of the wave recovers exactly the spatial shape of the soliton. Thus, temporal and spatial derivatives are equivalent up to a simple scaling as shown in Eqs. (1)–(4).

Orbits within the phase space of traveling wave dynamical systems, as defined by their state definition, correspond to waveforms of the governing PDE. In general, traveling waves are classified as *pulses*, *fronts*, or *wavetrains*. In the phase space, these waveforms correspond to distinct orbits. *Pulses* begin and end at the same fixed point, a homoclinic orbit (Fig. 2). *Fronts* are heteroclinic orbits that connect two fixed points (Fig. 3). Last, *wavetrains* form periodic orbits within the phase space, a limit cycle (Fig. 7). Thus, the goal of the proposed methodology is to provide an estimate of the vector field that yields the observed trajectories in the system's phase space.

III. METHODOLOGY

Sought are approximate autonomous traveling wave dynamical systems for one-way wave propagation in one spatial dimension with periodic or pseudo-infinite boundary conditions. The focus is on nonlinear waves upon saturation, that is, after one or multiple waves have formed and have reached their long-term shape and speed. This restriction in scope is appropriate as many nonlinear traveling waves in nature are observed after initial transients and saturation have occurred. Such dynamical systems are of the form

$$\frac{d}{d\xi}\mathbf{y}(\xi) = \mathbf{f}\left(\mathbf{y}(\xi)\right); \quad \mathbf{y}(\xi) \in \mathbb{R}^d, \quad \mathbf{f} : \mathbb{R}^d \to \mathbb{R}^d, \qquad (5)$$

where $\mathbf{y}(\xi)$ is the d-dimensional state vector of the dynamical system evolving though the traveling wave coordinate $\xi = x - ct + a$ and $\mathbf{f}$ is a vector-valued function to be approximated.

For the specific task of modeling traveling waves from data, the step of discovering the governing PDE is superficial. While discovery of the governing PDE may lend some interpretability to a system's behavior, such as its' dominant balance physics,[20] examination of governing equations at face value does little to inform the behavior of traveling waves or other coherent structures. If successfully discovered, to perform analysis tasks, one must either (i) simulate the PDE or (ii) seek traveling wave solutions. In this study, the step of discovering the governing PDE is skipped in favor of directly extracting traveling wave dynamical systems from data.

With the goal of producing a model of the form in Eq. (5), the methodology is presented in three sections: (i) the construction of the state of the observed system, (ii) the coordinate transformation of system observations, and (ii) the regression technique.

A. State construction

Assumptions regarding the mechanism(s) of wave propagation must be made to properly construct the state $\mathbf{y}$ and constrain the model to be a set of coupled first-order ODEs. Commonly encountered is diffusion-enabled wave propagation. Higher-order spatial derivatives associated with diffusion or hyper-diffusion result in differentiation in ξ of the same order, as shown in Eqs. (3) and (4). Therefore, if the wave propagation mechanism is assumed to be via diffusion, measurements or observations of the diffusing quantity are supplemented with their *spatial* derivative. For example, for single-component (U, for example) diffusion, the corresponding state vector is

$$\mathbf{y} = \begin{bmatrix} U \\ U' \end{bmatrix} = \begin{bmatrix} U \\ V \end{bmatrix}, \qquad (6)$$

where the definition $V = U'$ ensures that the system remains first order.

Diffusion-enabled propagation can generally be inferred by gradual transitions at the interface(s) between a wave and the quiescent medium. Exclusively advection-based propagation, usually marked by sharp fronts, discontinuities, or shocks, remains first-order after the coordinate transformation into ξ. More commonly encountered are advection-diffusion systems whose measurements of the diffusing quantities would need to be supplemented with corresponding spatial derivatives.

B. Data transformation

Transforming observed spatiotemporal dynamics into the traveling wave coordinate is required to perform the proposed modeling tasks. Typically, this transformation requires knowledge of the wave speed c and spatial offset a such that ξ is uniquely defined for an entire spatiotemporal field. However, when visualized in the phase space, all system snapshots of steadily propagating waves reside on trajectories (homoclinic orbits, heteroclinic orbits, or limit cycles) described by the underlying traveling wave ODEs. Thus, only a single snapshot of the waveform is required to completely define the trajectory of the traveling wave dynamical system through the phase space. This is shown explicitly in Fig. 2 where multiple snapshots of a traveling KdV soliton are shown in the phase space of the system's traveling wave dynamical system. Recall that the considered systems are autonomous; i.e., there is no explicit dependence on ξ. If each ith snapshot is treated as the system's initial condition—the state at a newly defined time $\tau_i = 0$—the coordinate transformation can be simplified by mapping each snapshot uniquely through $(x, t_i) \rightarrow (\xi_i = x - (c \cdot \tau_i) + a = x + a)$. Furthermore, the spatial offset a can be chosen to be zero; thus, each snapshot can be uniquely mapped to a new coordinate $(x, t_i) \rightarrow (\xi_i = x)$. This particular choice for the snapshot coordinate transformation can be exploited algorithmically in that (i) the speed of the wave need not be known and (ii) no data manipulation is required to perform the transformation. If necessary, multiple snapshots of the traveling wave can be phase-averaged to reduce the corrupting influence of noise or to artificially create a higher-resolution waveform (via smoothing splines or similar).

Given a single system snapshot for the steady wave case, the proposed methodology reduces to the procedure of performing established dynamical systems modeling techniques along the *spatial* direction of spatiotemporal data (as opposed to the temporal direction). Stressed is that shifting reference frames or other input data manipulation is not necessary nor is supplying the traveling wave speed to the algorithms.

C. Regression with neural ODEs

NODEs continuously evolve a system state $\mathbf{z}$, the dependent variable, through an independent variable (usually time, though ξ in this context), according to input–output mapping as performed by a neural network NN,

$$\mathbf{z}(\xi_1) = \mathbf{z}(\xi_0) + \int_{\xi_0}^{\xi_1} \mathrm{NN}(\mathbf{z}(\xi); \theta) d\xi, \qquad (7)$$

where θ are the parameters associated with the neural network NN. In general, the system defined by Eq. (7) can be non-autonomous. However, all systems presented in this paper are autonomous. Furthermore, although the integrand in Eq. (7) is a neural network, in general, the integrand can assume other functional forms. For example, a vector-valued function can be integrated where one component is an analytic function and another is a trainable neural network. For a given NN and θ, evaluating Eq. (7) amounts to solving the Initial Value Problem (IVP) with a standard ODE solver. For this paper, the notation ODESolve() is used to designate the solution of the IVP,

$$\mathbf{z}(\xi_1) = \mathrm{ODESolve}(\mathbf{z}(\xi_0), \mathrm{NN}(\mathbf{z}(\xi); \theta), \xi_0, \xi_1). \qquad (8)$$

Model residuals are backpropagated through the solution of the IVP, thereby obtaining the gradients of the loss function with respect to the neural network parameters necessary to select the parameters θ of the neural network NN.

For the present work, all examples considered have a uniform spatial discretization. Loss functions are taken to be a comparative metric (typically Euclidean distance) between observed trajectories, $\mathbf{y}(\xi_i) = \mathbf{y}_i$, and model trajectories $\mathbf{z}(\xi)$ evaluated at the uniformly spaced evaluation points. Given a grid spacing of $\Delta\xi$, the discretized output data stream is defined as

$$\mathbf{z}_i = \text{ODESolve}(\mathbf{z}(\xi_0), \text{NN}(\mathbf{z}(\xi);\theta), \xi_0, \xi_{end})\big|_{\xi=i\Delta\xi}, \tag{9}$$

that is, the sequence of evaluations of the solution to the IVP at points co-located with the input data stream. The distance between the two discretized trajectories is

$$\mathscr{L} = \sum_i \left(\mathbf{y}_i - \mathbf{z}_i\right)^2. \tag{10}$$

Open source implementations of NODEs are available in Python and Julia. The examples in this paper are implemented in the Julia ecosystem with the DifferentialEquations.jl[21] and DiffEqFlux.jl[16] packages. For details regarding the algorithmic implementation of neural ODEs, extensive documentation is provided in Chen *et al.*[15] and Rackauckas *et al.*[16]

IV. EXAMPLES

Examples in this section are organized with increasing complexity, beginning with synthetic data of simple monotonic fronts of a reaction–diffusion system and ending with modeling experimentally obtained wave profiles from a rotating detonation engine. Emphasis is placed on application and implementation over overall accuracy; thus, all examples are constructed with the same neural network architecture and without regularization. A summary of the examples provided in this paper is given in Table I.

A. Fisher–KPP fronts

The Fisher–KPP equation is a canonical mathematical system for modeling reaction–diffusion phenomena in a variety of systems, including combustion, chemistry, and ecology.[22] The nondimensionalized equation reads

$$u_t = u_{xx} + u(1-u). \tag{11}$$

Equation (11) has steady solutions: one corresponding to $u(x,t) = 0$ and one corresponding to $u(x,t) = 1$. From inspection, one can expect values of $0 < u < 1$ to tend toward the $u(x,t) = 1$ state. Figure 3 shows the developed steady traveling wave front simulated on a domain $x \in [0, 50]$ with grid spacing $\Delta x = 0.5$ using the method of lines. The vertical cut shown in Fig. 3(a) corresponds to the wave front profile of Fig. 3(b) and the phase portrait of Fig. 3(c).

TABLE I. Example traveling wave systems.

Section	System	Highlighted property
IV A	KPP–Fisher	Monotonic fronts
IV B	KdV–Burgers	Dispersive–dissipative shock fronts
IV C	FitzHugh–Nagumo	Wavetrains and fast/slow dynamics
IV D	RDE	Experimental data and state augmentation

Substitution of the TWA into Eq. (11) yields the coupled ODE,

$$\begin{cases} U' = V, \\ V' = -cV - U(1-U), \end{cases} \tag{12}$$

where $c = 2$, the speed of the front of the wave. The traveling wave dynamical system possesses fixed points at $(U, V) = (0, 0)$ and at $(U, V) = (1, 0)$. Wave front solutions connect these two fixed points in a heteroclinic orbit. The V-component of the vector field defined in Eq. (12) is shown in Fig. 4(a).

The single trajectory shown in Fig. 3(b) is used as the training data for application of the proposed method. Assuming this single trajectory is the only knowledge of the system, one can infer that the system is reaction–diffusion in type (as evidenced by diffusive appearance of the wave front). Thus, the state is assumed to be fully defined by the coordinate $(U, U') = (U, V)$. Observations of the variable U are, therefore, supplemented by their numerically computed spatial derivative. For this example, a neural ODE is trained to approximate the traveling wave dynamical system with this diffusion-enabled propagation assumption embedded by

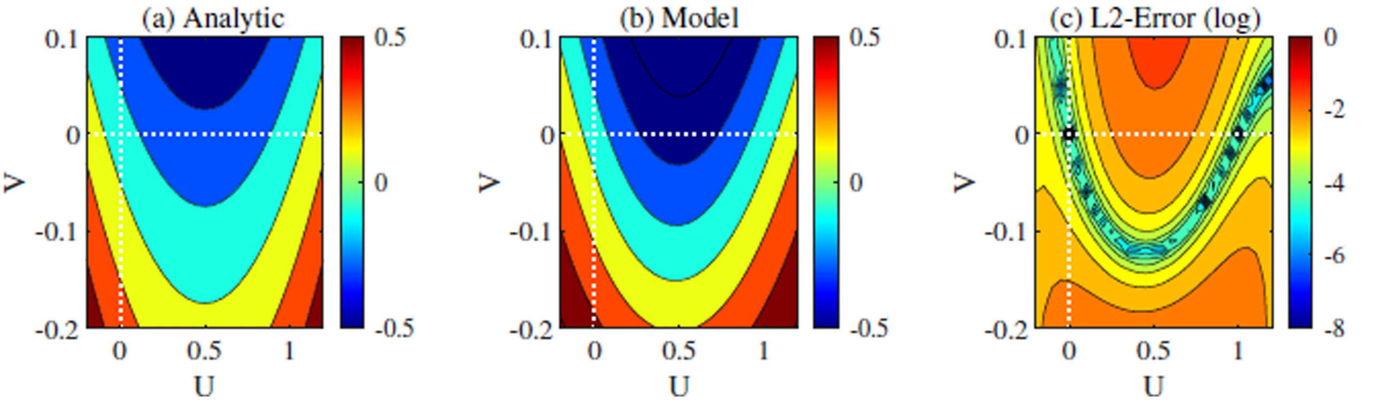


FIG. 4. Visualization of the V-component of the vector field corresponding to the KPP–Fisher traveling wave equations and the trained surrogate neural ODE model. (a) corresponds to the analytic vector field and (b) is the trained surrogate model. The $L2$ difference between (a) and (b) is shown in (c).

construction,

$$\begin{cases} U' = V, \\ V' = \mathrm{NN}(U, V; \theta). \end{cases} \tag{13}$$

The vector-valued function to be integrated is, therefore, defined as $\mathbf{g}(U, V; \theta) = (V, \mathrm{NN}(U, V; \theta))$. The loss function to be minimized, $\mathscr{L}$, is the Euclidean distance between the training trajectory (the $t = 15$ snapshot) and the model trajectory, $\mathbf{z}_i$, at the equally spaced solution points i,

$$\mathbf{z}_i = \mathrm{ODESolve}(\mathbf{y}_0, \mathbf{g}, 0, \xi_{end})\big|_{\xi = i\Delta\xi}, \tag{14a}$$

$$\mathscr{L} = \sum_i (\mathbf{y}_i - \mathbf{z}_i)^2, \tag{14b}$$

where θ are the neural network parameters to be optimized.

The initial condition chosen is the point $(U, V) = (0.9978, 9.65 \times 10^{-4})$, which is very near the fixed point $(1, 0)$. The lengths of the trained and evaluated trajectories are 75 grid points or a distance of 37.5. The neural network architecture used consists of one fully connected hidden layer of size 3 that is sigmoid-activated and a linear output layer. The ODE solver used for training and evaluation is a fifth-order Runge–Kutta integrator with a fourth-order interpolant. The BFGS optimization routine was used to minimize the loss to a value below 1×10^{-5}. Figure 4 shows the resultant vector field of the surrogate model compared with the analytic ground-truth. The wave front is successfully reconstructed to high accuracy. Although the surrogate model was only trained on the single steady wave front trajectory, the model is at least first order accurate for the region of the phase space shown in Fig. 4.

B. KdV–Burgers dispersive–dissipative shock wave

The viscous Burgers' equation possesses a prototypical *shock structure solution* connecting pre- and post-shock states that can be analytically solved for upon substitution of the TWA. Regularization with a dispersive term results in the KdV–Burgers equation,[23] given as

$$u_t + uu_x + u_{xxx} = \nu u_{xx}. \tag{15}$$

For certain choices of viscosity (ν), the KdV–Burgers equation admits a steadily propagating front connecting upstream and downstream states via an oscillatory front. This frontal structure is a prototypical example of a *dispersive–dissipative* shock. Numerical integration of the PDE on the domain $x \in [-100, 300]$ with $\Delta x = 0.5$ is shown in Fig. 5(a), with the wave profile displayed in Fig. 5(b) and the corresponding trjectory through the phase space in Fig. 5(c). The initial condition for the simulation is $u(x, t = 0) = \frac{1}{2}\left(1 + \tanh(x)\right)$. Viscosity is $\nu = 0.05$. Substitution of the TWA into Eq. (15) yields the coupled ODEs,

$$\begin{cases} U' = V, \\ V' = -\frac{1}{2}U(U - 1) + \nu V. \end{cases} \tag{16}$$

For the value of viscosity chosen, $\nu = 0.05$, the fixed points of the system exist at $(U, V) = (0, 0)$ and $(U, V) = (1, 0)$. The origin is a saddle point. The coordinate $(1, 0)$ is an attracting spiral—this is clearly observed in Fig. 5(c).

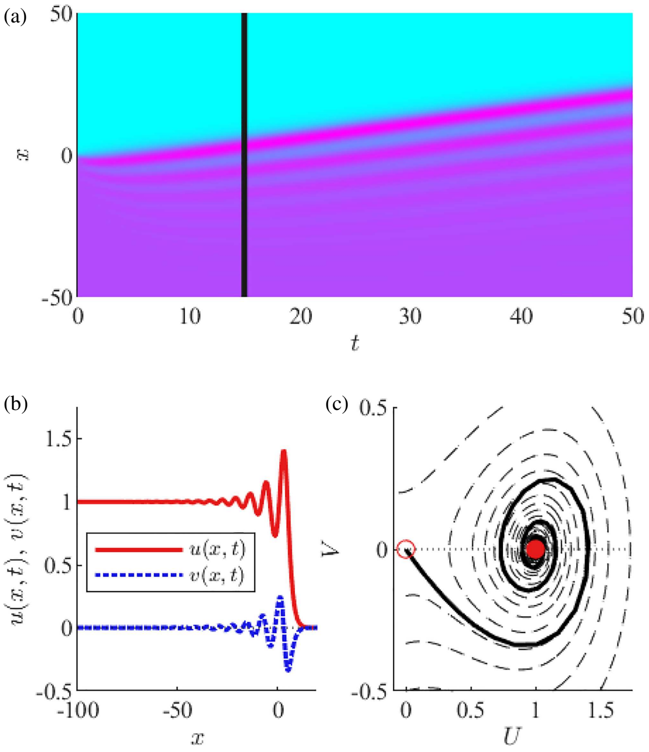


FIG. 5. (a) Solution of the KdV–Burgers PDE on the closed domain $x \in [-100, 300]$. (b) The 1D waveform at time $t = 15$. This waveform is the training set for the neural ODE. (c) The phase space representation of the training data waveform.

From inspection of the waveform in Fig. 5(b), the assumption of diffusion-enabled propagation is made for this system. Therefore, to properly construct the state, measurements of the variable U are supplemented with their numerically computed spatial derivatives. The state is the coordinate $(U, U') = (U, V)$. The diffusion-enabled propagation assumption is embedded by construction in the same manner as in Eq. (13),

$$\begin{cases} U' = V, \\ V' = \mathrm{NN}(U, V; \theta). \end{cases} \tag{17}$$

The vector-valued function to be integrated is, therefore, $\mathbf{g}(U, V; \theta) = (V, \mathrm{NN}(U, V; \theta))$. The loss function is the Euclidean distance between the training and evaluation trajectories at the uniformly spaced solution points i,

$$\mathbf{z}_i = \mathrm{ODESolve}(\mathbf{y}_0, \mathbf{g}, 0, \xi_{end})\big|_{\xi = i\Delta\xi}, \tag{18a}$$

$$\mathscr{L} = \sum_i (\mathbf{y}_i - \mathbf{z}_i)^2. \tag{18b}$$

The neural network architecture consists of a single sigmoid-activated hidden layer of size 3 and a linear output layer. The initial condition used in training is the coordinate $(U, V) = (5.07 \times 10^{-3}, -3.63 \times 10^{-3})$, which is near the fixed point $(0, 0)$. The

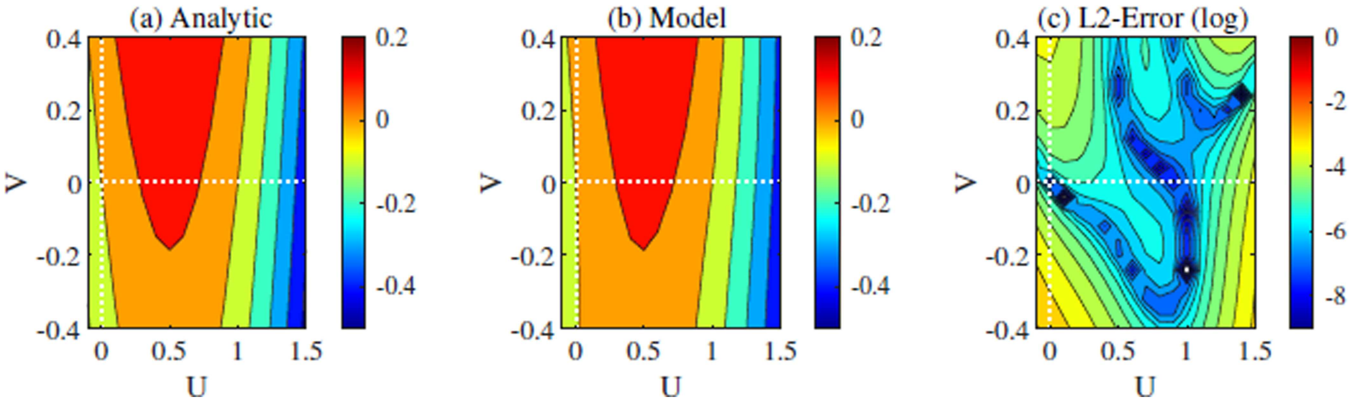


FIG. 6. Visualization of the *V*-component of the vector field corresponding to the KdV–Burgers traveling wave equations and the trained surrogate neural ODE model. (a) corresponds to the analytic vector field, and (b) is the trained surrogate model. The *L*2 difference between (a) and (b) is shown in (c).

lengths of trained and evaluated trajectories are 256 grid points or a distance of 128. This trajectory segment represents the frontal dynamics only; steady states are excluded from training. The ODE solver used in training and evaluation is a fifth-order Runge–Kutta integrator with fourth-order interpolant. The BFGS optimization routine was used to minimize the loss to a value below 1×10^{-5}.

Comparisons of the trained surrogate model and the analytic *V*-component of the vector field are provided in Fig. 6. For the single trajectory training set, the surrogate model is approximately second-order accurate for the entirety of the phase space shown in Fig. 6.

C. FitzHugh–Nagumo wavetrain

The 1D diffusive Fitz–Hugh Nagumo (FHN) model is a simple mathematical model that inherits the features of reactivity, spiking, and refractoriness from the Hodgkin–Huxley model for neuronal dynamics.[24] The model is given by

$$\begin{cases} u_t = \nu u_{xx} - v + u(1-u)(u-a), \\ v_t = \varepsilon(bu - v), \end{cases} \tag{19}$$

where $u(x,t)$ is the voltage in an axon and $v(x,t)$ is a slowly evolving recovery variable. In Eq. (19), only the voltage is diffusive, though in general, both the recovery variable and the voltage may diffuse and at different rates. The dynamics of the voltage follow a cubic nonlinearity that mimics reactivity; upon reaching a threshold voltage, a, the reaction term $u(1-u)(u-a)$ is positive and causes a spike in the dynamics. Depending on parameter values chosen, the FHN system can exhibit interesting dynamics such as solitary pulse propagation and the development of pulse trains.

For the present example, a surrogate model for a FHN pulse train is trained and analyzed with the goal of identifying the cubic nonlinearity. Numerical integration of the model in Eq. (19) was performed via the method of lines for $\varepsilon = 0.05$, $\nu = 1.0$, $a = -0.1$, and $b = 0.3$ on the periodic domain $x \in [0, 500)$ with unit grid spacing. Beginning with an initial condition of $u(x,0) = \frac{1}{2} + \frac{1}{2}\sin\left(\frac{10\pi}{L}x\right)$ and $v(x,0) = \frac{1}{2} + \frac{1}{2}\cos\left(\frac{10\pi}{L}x\right)$, the dynamics quickly settle into a pulse train with five distinct pulses that travel at a speed of $c = 1.11$. The spatiotemporal evolution of $u(x,t)$ and the steady waveform are shown in Fig. 7.

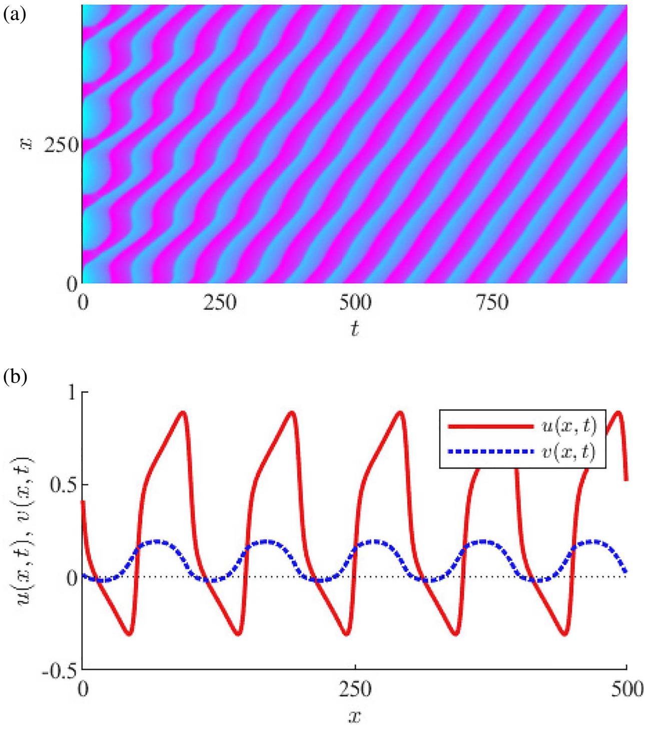


FIG. 7. In (a), displayed is a simulation of the FitzHugh–Nagumo equations on a periodic domain of length $L = 500$ starting from a 5-peaked sinusoidal initial condition. In (b), the steady-state waveform is shown.

Substitution of the TWA into Eq. (19) yields the three coupled ODEs,

$$\begin{cases} U' = W, \\ V' = \frac{\varepsilon}{c}\left(bU - V\right), \\ W' = \frac{1}{\nu}\left(-cW + V - U\left(1-U\right)\left(U-a\right)\right). \end{cases} \tag{20}$$

These three-dimensional dynamics are examined on the plane $W = 0$, which exposes the structure of the nonlinearity embedded in W'. Two nullclines exist on this plane defining where $V' = W' = 0$. The nullcline $W' = 0$ corresponds to the cubic polynomial describing the excitability or reactivity of the medium; $V = U(1-U)(U-a)$ (see Fig. 9).

A neural ODE is trained against the single wavetrain profile shown in Fig. 7. The propagation mechanism is assumed to be via diffusion. The state of the system is given by the coordinates $(U, U', V) = (U, W, V)$. The neural ODE is, therefore, constructed with the constraint that $U' = W$,

$$\begin{cases} U' = W, \\ V' = \mathrm{NN}\left(U, V, W; \theta\right)_1, \\ W' = \mathrm{NN}\left(U, V, W; \theta\right)_2. \end{cases} \tag{21}$$

The vector-valued function to be integrated is, therefore, $\mathbf{g}(U, V, W; \theta) = (W, \mathrm{NN}_1, \mathrm{NN}_2)$, where NN_1 and NN_2 designate the first and second components of the neural network output. The neural ODE is trained via *minibatching* in order to avoid issues in obtaining trivial local minima. The objective function is

$$\mathscr{L} = \sum_{i=1}^{N/k}\left(\mathbf{y}_{ki+j} - \mathrm{ODESolve}(\mathbf{y}_{ki}, \mathbf{g}, 0, j\Delta\xi)\big|_{\xi=j\Delta\xi}\right)^2, \tag{22}$$

where N is the total number of data points along the trajectory, $k \in \mathbb{Z}^+$ is the degree of sub-sampling (every kth point along the training trajectory is used to seed an ODESolve), and $j \in \mathbb{Z}^+$ sets where along the trajectories the ODESolve is evaluated. For this

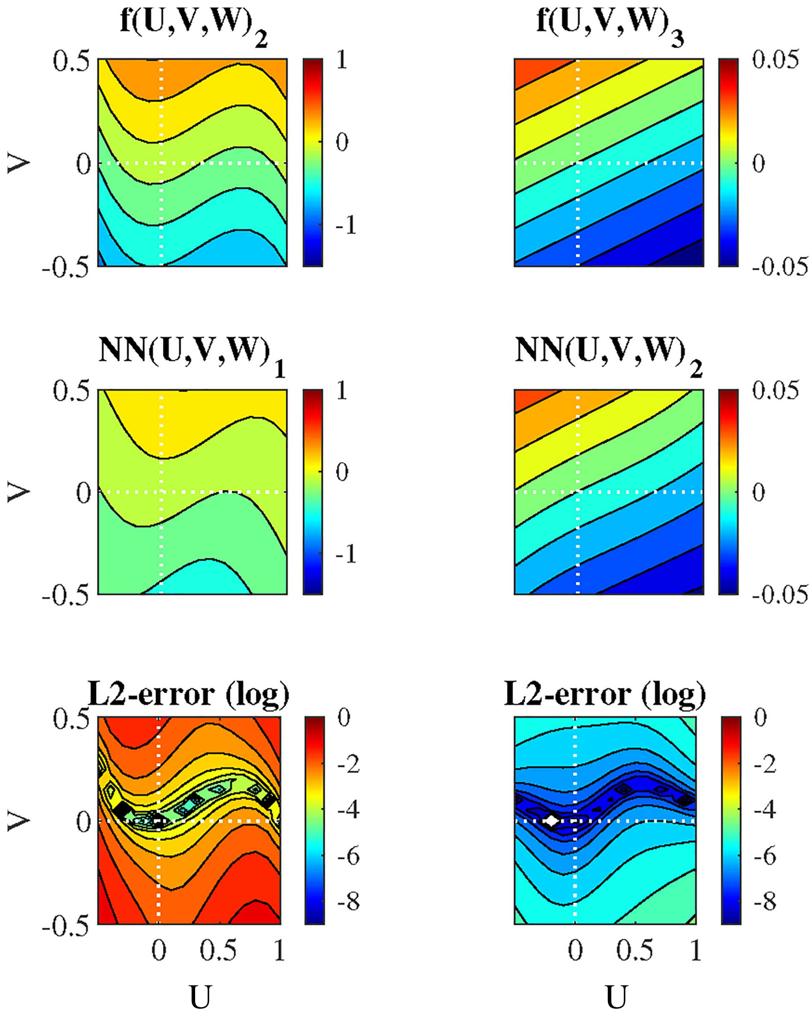


FIG. 8. Visualization of the vector field for the FHN traveling wave equations and trained surrogate neural ODE model on the plane $W = 0$. The traveling wave ODE [Eq. (20)] second (cubic reaction nonlinearity) and third (linear recovery) components are displayed in top row, respectively. The corresponding surrogate model components are displayed in the middle row. The $L2$ difference between the analytic and model vector field components is given in the bottom row.

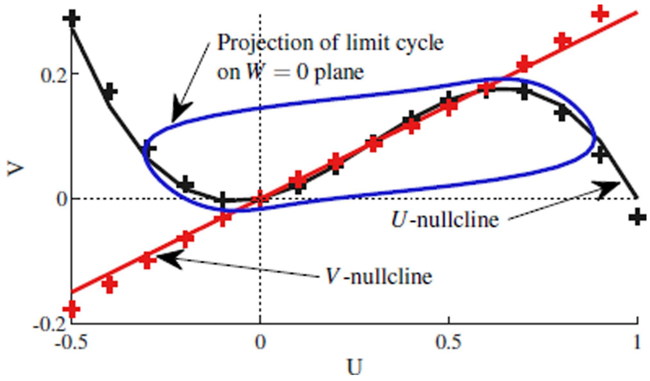


FIG. 9. *U*- and *V*-nullclines of the FHN traveling wave dynamical system and the trained surrogate neural ODE. The *U*-nullcline (black) corresponds to the cubic reaction term of the governing PDE. The *V*-nullcline (red) corresponds to the linear slow return of the recovery variable. Solid lines correspond to the analytic nullclines. Plot markers correspond to the numerically interpolated values from the neural ODE.

example, $k = 4$ and $j = 5$. This method exchanges the difficulty associated with periodic training data (such as arriving at trivial local minima) for a N/k-fold increase in ODE solves. Although this training process is undoubtedly slower, for oscillatory and stiff problems, this method of training is more robust. The neural network architecture consists of a single sigmoid-activated hidden layer of size 3 and a linear output layer. The ODE solver used for training and evaluation is an automatic stiffness-detecting solver that selects between a fifth-order Runge–Kutta integrator and a second-order Rosenbrock integrator. The BFGS optimization routine was used to minimize the loss to a value below 1×10^{-5}.

Figure 8 compares representative 2D vector fields on the plane $W = 0$ for the surrogate model and the analytic traveling wave ODEs. The numerically interpolated nullclines $V' = W' = 0$ for both systems are shown in Fig. 9. The solid lines correspond to the analytic nullclines; the plot markers correspond to the interpolated locations from the surrogate model. The analytic $W' = 0$ nullcline and the cubic polynomial fit of that of the surrogate model are

$$V_{true}(U) = -U^3 + 0.865U^2 + 0.135U, \tag{23a}$$

$$V_{model}(U) = -1.06U^3 + 0.907U^2 + 0.125U + 0.00162. \tag{23b}$$

Thus, the underlying cubic linearity of the system has been successfully approximated.

D. Data-driven modeling of rotating detonation waves

The Rotating Detonation Engine (RDE) is a novel thrust-producing combustion engine whose defining attributes are (i) the promotion of constant volume heat release,[25] (ii) mechanical simplicity, and (iii) high power density.[26] Most laboratory-grade RDEs are constructed via concentric cylinders to form an annular combustion chamber. Separate injected streams of gaseous fuel and oxidizer enter the head end of the device where they rapidly mix and flow downstream. Ignition is provided via a torch, pre-detonator, or automotive spark plug. Once chemical reactions begin, heat is rapidly released into the combustion chamber, forming strong gradients in the density and temperature of the fluid. Because of the geometric confinement and nature of the periodicity of the combustion chamber, these gradients often self-steepen into wave fronts coupled to regions of intense chemical reactions. The waves travel about the annulus consuming the newly injected and mixed fluid ahead of their path, so long as enough chemical potential energy exists within the mixture to sustain the wave.[27] These waves differ from those of classical combustion wave theory in that they are subject to dissipative effects, such as heat transfer to the walls of the device[28] and rapid lateral expansion downstream.[29,30] These physics give the RDE flowfield its characteristic “sawtooth” waveform, with repeatable flow patterns and layer stratification.[31] A notional RDE and a two-wave flowfield is shown in Fig. 10.

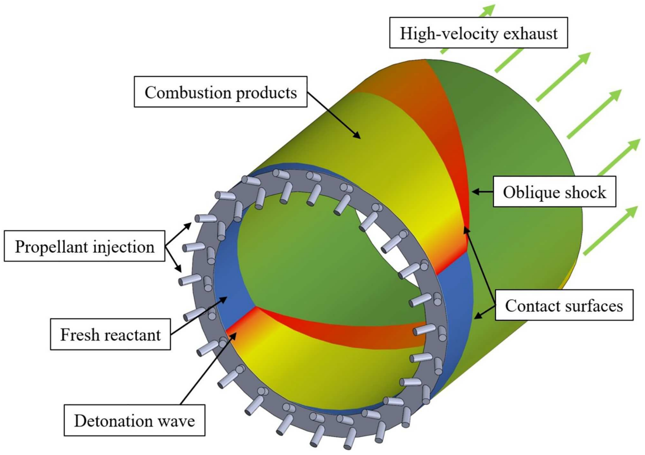


FIG. 10. Notional rotating detonation engine with major features labeled. In steady operation, a number of detonation waves circumnavigate a periodic combustion chamber indefinitely, consuming a newly injected propellant in the paths of the waves.

The behavior of the combustion waves, including the coupling mechanisms with engine sub-components (injectors, mixing scheme, exhaust, etc.), while intuitively straightforward, is exceptionally difficult to model.[32,33] The physics are fundamentally multi-scale. The combustion physics can occur over length scales $\ll 10^3$ m and can travel supersonically, in some cases over kilometers per second.[34] In the other extreme, the time scale associated with a fluid particle moving from the propellant plenums to the exhaust plane of the device may be several orders of magnitude slower than that of the combustion physics. The scales of mixing may be similarly slow in time and long in length.[35]

These scales are intimately coupled: sufficient propellant injection, mixing, and exhaust rejection must occur within the transit time of the combustion waves. Thus, the overall behavior of the device is fundamentally the result of physics coupling across these widely varied scales. Should these scales be altered, a variety of transient behavior and operating modes have been observed in the experimental and computational literature. These behaviors include mode-locking,[36] varied wave counts,[37] bifurcations of wave counts,[36,38] modulation, and “breathing” plane waves.[36] Modeling

RDE physics is therefore often constrained to device-specific geometries and operating conditions. Furthermore, such simulations need to capture the cascade of physical scales such that their interactions are captured and fully developed. Such models are computationally expensive and limited to a small set of operating conditions, severely limiting the task of physics investigation. Recent advances in mathematical modeling have sought to reduce the fundamental physical processes into low-order evolution equations with artificially imposed physical scales.[36] These ideas have been extended to the single-dimension Euler equations for an inviscid, compressible flow on a periodic domain.[39] However, matching the relevant scales of an specific experimental apparatus in a tractable model is a difficult task; all physical scales and their interactions must be sufficiently accounted for. In the RDE, with an exceptionally harsh (high temperature and pressure) sensing environment, high-quality and well-resolved spatial and temporal measurements of the flow state are usually unattainable. Many RDEs in the laboratory setting are therefore monitored with high-speed cameras positioned downstream of the combustor to image the aft-end of the device. Captured images clearly contain wave kinematic properties. Wave kinematics can then be correlated to engine operation, including stability and performance.[39]

For the present example, a surrogate model of a RDE running at a specific operating point is constructed directly from experimentally obtained high-speed camera footage. The goal of this exploratory model is to provide estimates for time and spatial scales of the combustion wave fronts present in the combustor. The experimental apparatus is a 76-mm flowpath outside a diameter engine with a flowpath length of 76-mm. Fuel (methane gas) and oxidizer (oxygen gas) are fed into the combustion chamber through a set of discrete injectors equally spaced around the engine annulus. Complete details of the experimental apparatus can be found in Ref. 36. For this example, the engine was run with a global equivalence ratio of 0.32 and a mass flow rate of 226 g/s. The engine firing was monitored with a high-speed camera recording images at 240 000 frames/s with a resolution of 128 × 128 pixels. The high-speed camera footage is typically post-processed to yield a space-time history of the waves for the duration of the experiment. A post-processed half-millisecond window of operation for this particular experiment is shown in Fig. 11(a). In steady operation, four waves traveling stably with a measured speed of approximately $c = 1250$ m/s were observed. The vertical white line in Fig. 11 corresponds to the waveform shown in Fig. 11(b) (after mean-subtracting the data). Because the waves are approximately steady, a phase-averaged profile can be readily obtained, as shown in Fig. 11(c) for one period of the wave. Note that in this example, the waves are left-running. In general, rotating detonation waves can travel left, right, or in both directions, creating standing wave patterns.

The relationship between pixel-integrated luminosity (such as what is shown in Fig. 11) and flow quantities of interest (pressure, temperature, species, etc.) is unknown except for the qualitative knowledge that more luminosity generally implies a more intense region of combustion. However, luminosity as captured by high-speed camera is spatially and temporally resolved. Furthermore, the waveform is qualitatively similar to the steep-fronted waves typical of many canonical reaction–diffusion–advection or combustion systems. A surrogate model is constructed from this observable to provide estimates of the scales present in this particular experiment.

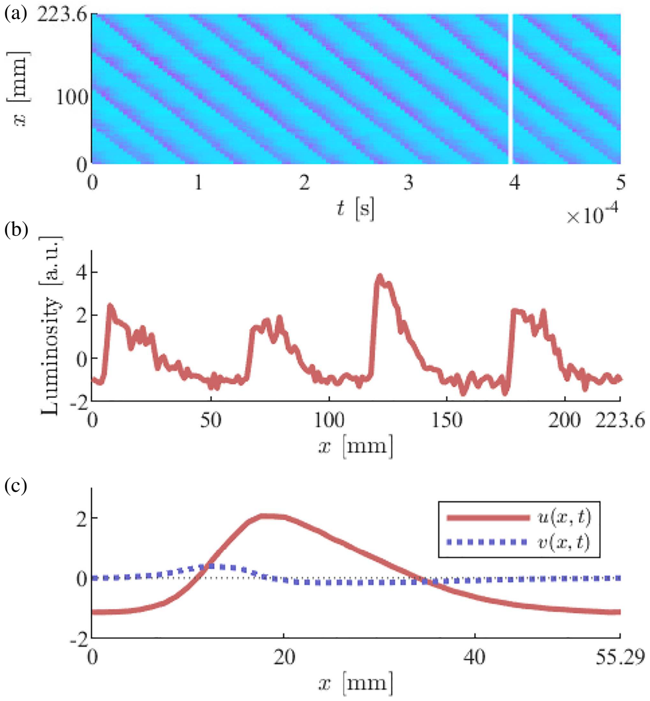


FIG. 11. A common monitoring technique used for rotating detonation engines is through high-speed camera imaging. In (a), a half-millisecond window of operation in a rotating detonation engine experiment is shown (after performing an intermediate post-processing step arranging the camera frames into 1D system snapshot vectors). In this experiment, four waves travel about the periodic combustion chamber with a speed of approximately 1250 m/s. In (b), a snapshot in time of pixel-integrated luminosity from the high-speed camera is shown, giving a distinctive sawtooth-like waveform. Because the waves are approximately steady, the system snapshots can be phase-averaged to produce a representative single-wavelength shape shown in (c).

To construct the surrogate model, the observed waves are assumed to be reaction–diffusion–advection in type. Thus, the measurements of luminosity $[u(x,t) = U(\xi)]$ are supplemented with their spatial derivative $[v(x,t) = U'(\xi) = V(\xi)]$. Additionally, known is that the system is subject to the competing physics of combustion (depletion of fuel mixture) and injection and mixing (replenishment of fuel mixture), which occur on different physical scales. In an attempt to capture these physics, the state is further augmented with a third state variable $[w(x,t) = W(\xi)]$ whose interactions with the remaining state variables can allow for the desired activator-inhibitor type behavior. The surrogate model becomes the coupled set of ODEs,

$$\begin{cases} U' = V, \\ V' = \mathrm{NN}\,(U, V, W; \theta)_1, \\ W' = \mathrm{NN}\,(U, V, W; \theta)_2. \end{cases} \tag{24}$$

The vector-valued function to be integrated is therefore $\mathbf{g}(U, V, W; \theta) = (V, \text{NN}_1, \text{NN}_2)$. The objective function used is based on Eq. (22) though modified to only penalize differences between the observable and its reconstruction, as opposed to each state variable,

$$\mathscr{L} = \sum_{i=1}^{N/k} \left(U_{ki+j} - \text{ODESolve}\left(\mathbf{y}_{ki}, \mathbf{g}, 0, j\Delta\xi\right)_1 \Big|_{\xi=j\Delta\xi} \right)^2. \qquad (25)$$

Because there are no measurements of or constraints on W, the initial condition $\mathbf{y}_{ki}$ passed to each ODESolve is ambiguous. The value of W at $\xi = 0$ is arbitrarily set to a constant reference value. For this example, $W(0) = -U(0)$. Each ODESolve in Eq. (25) references trajectories of W starting at this reference value. The initial condition in Eq. (25) is, therefore, defined recursively,

$$\mathbf{y}_{ki} = \begin{bmatrix} U_{ki} \\ V_{ki} \\ \text{ODESolve}\left(\mathbf{y}_{k(i-1)}, \mathbf{g}, 0, \Delta\xi\right)_3 \Big|_{\xi=\Delta\xi} \end{bmatrix}. \qquad (26)$$

For this example, the input data periodic domain is $x \in [0, 223.6)$ mm discretized into 180 data points. The minibatch parameters k and j are 1 and 7, respectively. The neural network architecture consists of a single sigmoid-activated hidden layer of size 3 and a linear output layer. The training data set is the phase-averaged single-wavelength profile of luminosity shown in Fig. 11(c). This segment of input data consists of 45 equally spaced data points spanning approximately 55.29 mm.

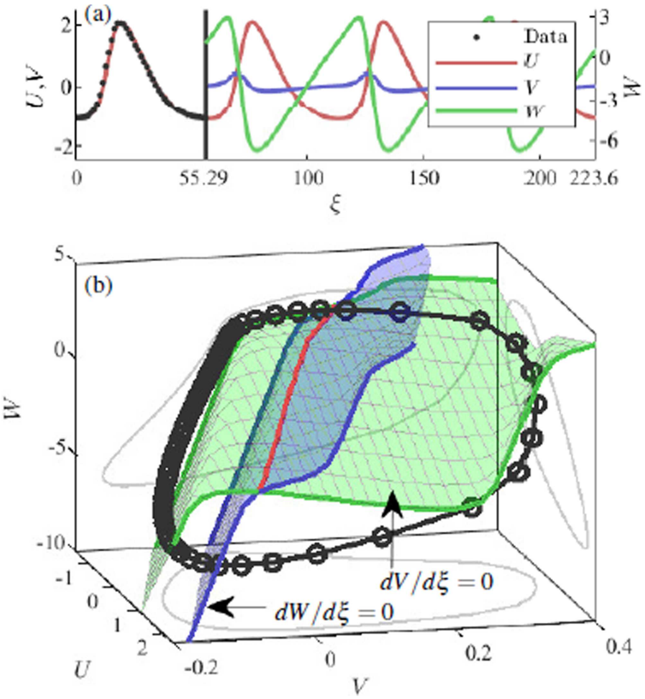


FIG. 12. In (a), the single-wavelength training data are shown as black dots in the region from $0 < \xi < 55.29$ mm. The reconstruction of the training data is shown in red. The prediction of the waveform beyond one period is shown to the right of the vertical black bar for each of the state variables. In (b), the predicted limit cycle is shown in the phase space. The green and blue surfaces correspond to locations in the vector field where $dV/d\xi = 0$ and $dW/d\xi = 0$, respectively. The intersection of these two surfaces is given by the red line. The predicted limit cycles show clear slow/fast behavior as the system state undergoes periods of fast reactions and slow replenishment of fuel mixture.

Figure 11(a) shows the phase-averaged luminosity training data to the left of the vertical black line. The black data points overlay the reconstruction of the luminosity profile via the trained neural ODE. To the right of the black line, each component of the neural ODE is shown for a prediction length of three additional periods. The trajectory for the augmented state variable W assumes a periodic waveform that exhibits clear slow/fast behavior. As luminosity increases, the value of W begins to rapidly decline until just after the peak in luminosity. Once at a minimum, the value of W slowly increases until the process repeats. In this way, the behaviors of U and W can be interpreted as those of an activator–inhibitor system interacting across their physical scales. The scales for reactions and regeneration of fuel mixture can be estimated from the space-time behavior of the inhibitor (W) and knowledge of the (observed) wave speed $c = 1250$ m/s. Reactions occur over a length of 12 mm corresponding to a time of 9.6×10^{-6} s. Regeneration of fuel mixture occurs over the remaining space in the wavelength; i.e., $55.29 - 12 = 43.29$ mm. This length corresponds to a time of 3.5×10^{-5} s. Note that the magnitude of W and its physical units is arbitrary.

The slow/fast dynamics exhibited in the waveforms can be further explored in the phase space. Figure 12(b) shows the predicted limit cycle in the three-dimensional phase space. Included in this figure are surfaces denoting locations in the phase space where $dV/d\xi = 0$ (green surface) and where $dW/d\xi = 0$ (blue surface). The intersection of these two surfaces is shown with a red line. The green surface $dV/d\xi = 0$ constitutes an estimate of the slow manifold of the system. System trajectories flow slowly along this surface until an increase in U sends the system on a fast round-trip excursion through the phase space. This separation of fast/slow physics is qualitatively similar to that of the FHN system explored in Sec. IV C.

V. DISCUSSION

Data-driven modeling of spatiotemporal dynamics relies upon using principled methods to (i) identify underlying system structure and (ii) leverage the found structure in such a way to make predictions or inferences regarding the system. Many data-driven techniques utilize the Proper Orthogonal Decomposition (POD) to identify and order spatiotemporal modes that contribute to observed system dynamics. This method and its derivatives have been used with great success for system diagnostics, physics investigations, and future state predictions. In this work, considered is a narrow scope of systems that admit 1D traveling waves. This known structure can be leveraged to directly model traveling wave dynamical systems from data without invoking the POD and inheriting its limitations, i.e., its inability to handle translations and rotations. The methodology described in this paper uses Neural Ordinary Differential Equations (NODEs) to best approximate a traveling wave dynamical system's phase space and system trajectory. Algorithmically, this regression amounts to training a system of NODEs to reproduce a system's *spatial* waveform as opposed to the *temporal* waveform. This methodology is advantageous because (i) the traveling wave ansatz $\xi = x - ct + a$ is satisfied by construction, (ii) the difficulty associated with calculating *time* derivatives for spatiotemporal data has been exchanged with calculating *spatial* derivatives, and (iii) only a single snapshot of a traveling wave profile is required to perform the regression (no knowledge of the speed of the waves is required).

In this section, highlighted are several areas where the methodology can be improved upon or modified. Last, discussed is the outlook for the application of these ideas to real-world physical systems.

A. Unsteady wave motion and multiple transport speeds

Many observed waves can both persist in time and not travel with constant velocities. Examples include shallow water waves with periodic bathymetry, periodically interacting solitons, and modulated rotating detonation waves. For small perturbations to wave speed, this methodology can still be applied to produce a surrogate model that can inherit important properties of the underlying system, such as the estimates of system's fixed points or invariant manifolds. However, large fluctuations or drifts in wave speeds will cause failures in convergence during training models or during model evaluation.

Similarly, some traveling wave dynamical systems can admit waves of different speeds. For example, Korteweg–de Vries solitons of different amplitudes and speeds can co-exist on a shared domain. In this case, the methodology cannot be successfully applied: assumed is a single constant transport speed for the entirety of the training data. Regression applied on subsets of the domain containing isolated waves is a remedy for this specific situation.

B. Regression techniques

All examples presented in this paper use Neural ODEs (NODEs) as the regression backend. Furthermore, each example used identical neural network architectures, i.e., an input layer (with a dimension equal to that of the system's state), a single hidden layer with three sigmoid-activated nodes, and a linear output layer. Two styles of objective functions were used in the examples. For heteroclinic and homoclinic orbits, the Euclidean distance between training data and the model output was minimized. For periodic orbits, the models were trained via minibatching by seeding several model evaluations along the training trajectory. The relative simplicity of the network architecture and the choice of training strategy were used to avoid issues with over-fitting and/or attaining a local minimum.

For robust implementation of the proposed methodology, regularization is a necessary next step. In the context of NODEs, regularization is a rapidly evolving topic, with many proposed methods in the literature. These techniques include penalizing higher-order derivatives along a solution trajectory,[40] randomly sampling the end time of the ODE during training,[41] and leveraging techniques from optimal transport theory.[42]

Neural ODEs were selected to perform the regressions exhibited in this work because of the low barrier to entry and the flexibility of the model architecture. Other methods exist that are a drop-in replacement for neural ODEs. Sparse Identification of Nonlinear Dynamics (SINDy) is one such method that can be applied to the examples in this paper. However, arriving at a high-quality model is intimately coupled with the choice of functions in the method's library, which may require expert knowledge or tuning.

C. Outlook and value

This modeling strategy presents interesting and high-value opportunities in the physical sciences. Consistent with the motivation for this work, discussed here are (i) surrogate modeling for combustion systems and (ii) parameter tuning and closure modeling.

1. Surrogate modeling in combustion systems

The backbone of modern combustion theory is the mathematical balance of flame speeds and conservation equations (mass, momentum, energy, and species) subject to heat transfer and other dissipative effects. A steady flame (e.g., a Bunsen burner or a candle flame) represents a configuration in which all of the involved physical processes balance in space and in time. Because flames are self-supported localized regions of combustion, these structures are often studied as propagating wave fronts. In laminar flame theory, the explicit wave structure is often exploited (the Shvab–Zeldovich transformation,[43] for example) to reformulate governing physical conservation laws—partial differential equations—as a set of coupled ODEs that are amenable to analysis. These from-first-principles approaches require detailed knowledge of diffusivities, conductivities, and chemical kinetics; these properties are difficult to obtain for real-world systems.

Because many flame configurations are either 1D or can be approximated as 1D in directions normal to the flame interface,[44] the methodology presented in this paper can be readily applied to these systems. Surrogate models can be constructed directly from flame observables much in the same way the RDE surrogate was constructed in Sec. IV D. In many laboratory settings, laser diagnostics can provide high spatial resolution snapshots of flowfield temperature and species distributions for a wide variety of burner configurations. Outside of the laboratory, self-supported reaction–diffusion systems exist that can be similarly approximated as 1D normal to the reaction interface. A topical example is a wildfire front where the aggregate chemical kinetic rates of organic material in a forest competes with heat transfer (conduction and radiation), inflow of oxidizer (wind direction and speeds), and other transport phenomena.[45]

2. Parameter tuning and closure modeling

For many systems that admit traveling wave solutions, the governing equations are known or phenomenological models have been proposed to sufficiently mimic observations. A common task is to estimate or tune model parameters to fit data. Similarly, in many systems, the physics or its mathematical representation may only be *partially* known, in which case the missing physics needs to be recovered. This is particularly relevant in combustion theory; creating effective chemical kinetic submodels requires large databases of carefully maintained reaction networks that are calibrated via laboratory experiments. These reaction submodels may not generalize to specific flame configurations, stoichiometry, or other operating condition. In these situations, the traveling wave ansatz can be leveraged to reformulate governing equations in terms of ODEs that evolve through the traveling wave coordinate ξ with unknown parameters to be tuned or with a neural network to be trained to "close" the model by approximating unknown terms.

VI. CONCLUSION

The coordinate transformation $\xi = x - ct + a$ is a powerful tool to recast wave equations as a set of ordinary differential equations. In this paper, this coordinate transformation was leveraged to perform system identification and surrogate modeling within this latent space of the governing equations. Algorithmically, this coordinate transformation is *implicit*; the presented methodology reduces to applying well-established time-series modeling techniques along the spatial dimension of a system snapshot. For the present study, neural ODEs were used for this task. These ideas were applied to several examples, including fronts of the KPP–Fisher and KdV–Burgers equations, pulse trains of the FHN equation, and finally rotating detonation waves. In addition to reproducing observed waveforms to high accuracy, the resulting models captured the structure embedded within the traveling wave ODEs, including location and type of fixed points, limit cycle behavior, and heteroclinic and homoclinic orbits.

ACKNOWLEDGMENTS

This work was supported by the U.S. Air Force Center of Excellence on Multi-Fidelity Modeling of Rocket Combustor Dynamics (Award No. FA9550-17-1-0195). The author thanks Professor Karen Willcox and Dr. Ramakanth Munipalli for their valuable insights and comments regarding this work. Experiments were performed at the University of Washington High Enthalpy Flow Laboratory.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

[1] P. Holmes, J. L. Lumley, G. Berkooz, and C. W. Rowley, *Turbulence, Coherent Structures, Dynamical Systems and Symmetry* (Cambridge University Press, 2012).
[2] M. Greiner, O. Mandel, T. W. Hänsch, and I. Bloch, "Collapse and revival of the matter wave field of a Bose–Einstein condensate," Nature **419**, 51–54 (2002).
[3] J. N. Kutz, "Mode-locked soliton lasers," SIAM Rev. **48**, 629–678 (2006).
[4] P. C. Bressloff, "Traveling waves and pulses in a one-dimensional network of excitable integrate-and-fire neurons," J. Math. Biol. **40**, 169–198 (2000).
[5] N. Akhmediev and A. Ankiewicz, *Dissipative Solitons: From Optics to Biology and Medicine* (Springer Science & Business Media, 2008), Vol. 751.
[6] Z. Feng, "Traveling wave behavior for a generalized Fisher equation," Chaos, Solitons Fractals **38**, 481–488 (2008).
[7] T. Yazaki, A. Iwata, T. Maekawa, and A. Tominaga, "Traveling wave thermoacoustic engine in a looped tube," Phys. Rev. Lett. **81**, 3128–3131 (1998).
[8] J. Mandel, L. S. Bennethum, J. D. Beezley, J. L. Coen, C. C. Douglas, M. Kim, and A. Vodacek, "A wildland fire model with data assimilation," Math. Comput. Simul. **79**, 584–606 (2008).
[9] S. L. Brunton and J. N. Kutz, *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control* (Cambridge University Press, 2019).
[10] G. Berkooz, P. Holmes, and J. L. Lumley, "The proper orthogonal decomposition in the analysis of turbulent flows," Annu. Rev. Fluid Mech. **25**, 539–575 (1993).
[11] M. Couplet, C. Basdevant, and P. Sagaut, "Calibrated reduced-order POD-Galerkin system for fluid flow modelling," J. Comput. Phys. **207**, 192–220 (2005).
[12] J. Reiss, P. Schulze, J. Sesterhenn, and V. Mehrmann, "The shifted proper orthogonal decomposition: A mode decomposition for multiple transport phenomena," SIAM J. Sci. Comput. **40**, A1322–A1344 (2018).
[13] A. Mendible, S. L. Brunton, A. Y. Aravkin, W. Lowrie, and J. N. Kutz, "Dimensionality reduction and reduced-order modeling for traveling wave physics," Theor. Comput. Fluid Dyn. **34**, 385–400 (2020).
[14] S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Discovering governing equations from data by sparse identification of nonlinear dynamical systems," Proc. Natl. Acad. Sci. U.S.A. **113**, 3932–3937 (2016).
[15] R. T. Q. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud, "Neural ordinary differential equations," in *Advances in Neural Information Processing Systems*, edited by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Curran Associates, Inc., 2018), Vol. 31, pp. 6571–6583.
[16] C. Rackauckas, Y. Ma, J. Martensen, C. Warner, K. Zubov, R. Supekar, D. Skinner, A. Ramadhan, and A. Edelman, "Universal differential equations for scientific machine learning," arXiv:2001.04385v3 (2020).
[17] S. H. Rudy, S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Data-driven discovery of partial differential equations," Sci. Adv. **3**, e1602614 (2017).
[18] M. Raissi, P. Perdikaris, and G. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," J. Comput. Phys. **378**, 686–707 (2019).
[19] M. Raissi and G. E. Karniadakis, "Hidden physics models: Machine learning of nonlinear partial differential equations," J. Comput. Phys. **357**, 125–141 (2018).
[20] J. L. Callaham, J. V. Koch, B. W. Brunton, J. N. Kutz, and S. L. Brunton, "Learning dominant physical processes with data-driven balance models," Nat. Commun. **12**, 1016 (2021).
[21] C. Rackauckas and Q. Nie, "DifferentialEquations.jl–A performant and feature-rich ecosystem for solving differential equations in Julia," J. Open Res. Softw. **5**, 15 (2017).
[22] R. A. Fisher, "The wave of advance of advantageous genes," Ann. Eugen. **7**, 355–369 (1937).
[23] R. Johnson, "A non-linear equation incorporating damping and dispersion," J. Fluid Mech. **42**, 49–60 (1970).
[24] R. FitzHugh, "Mathematical models of threshold phenomena in the nerve membrane," Bull. Math. Biophys. **17**, 257–278 (1955).
[25] P. Wolański, "Detonative propulsion," Proc. Combust. Inst. **34**, 125–158 (2013).
[26] R. Huff, M. D. Polanka, M. J. McClearn, F. R. Schauer, M. L. Fotia, and J. L. Hoke, "Design and operation of a radial rotating detonation engine," J. Propul. Power **35**, 1143–1150 (2019).
[27] P. Strakey, D. Ferguson, A. Sisler, and A. Nix, "Computationally quantifying loss mechanisms in a rotating detonation engine," in *54th AIAA Aerospace Sciences Meeting*, AIAA 2016-0900 (American Institute of Aeronautics and Astronautics, 2016).
[28] J. Braun, J. Sousa, and G. Paniagua, "Numerical assessment of the convective heat transfer in rotating detonation combustors using a reduced-order model," Appl. Sci. **8**, 893 (2018).
[29] K. Y. Cho, J. R. Codoni, B. A. Rankin, J. Hoke, and F. Schauer, "Effects of lateral relief of detonation in a thin channel," in *55th AIAA Aerospace Sciences Meeting*, AIAA 2017-0373 (American Institute of Aeronautics and Astronautics, 2017).
[30] M. Reynaud, F. Virot, and A. Chinnayya, "A computational study of the interaction of gaseous detonations with a compressible layer," Phys. Fluids **29**, 056101 (2017).
[31] R. T. Fievisohn and K. H. Yu, "Steady-state analysis of rotating detonation engine flowfields with the method of characteristics," J. Propul. Power **33**, 89–99 (2017).

[32] P. A. Cocks, A. T. Holley, and B. A. Rankin, "High fidelity simulations of a non-premixed rotating detonation engine," in *54th AIAA Aerospace Sciences Meeting*, AIAA 2016-0125 (American Institute of Aeronautics and Astronautics, 2016).

[33] T. Gaillard, D. Davidenko, and F. Dupoirieux, "Numerical simulation of a rotating detonation with a realistic injector designed for separate supply of gaseous hydrogen and oxygen," Acta Astronaut. **141**, 64–78 (2017).

[34] V. Anand, A. S. George, R. Driscoll, and E. Gutmark, "Investigation of rotating detonation combustor operation with H_2-air mixtures," Int. J. Hydrog. Energy **41**, 1281–1292 (2016).

[35] C. A. Nordeen, D. Schwer, F. Schauer, J. Hoke, T. Barber, and B. M. Cetegen, "Role of inlet reactant mixedness on the thermodynamic performance of a rotating detonation engine," Shock Waves **26**, 417–428 (2015).

[36] J. Koch, M. Kurosaka, C. Knowlen, and J. N. Kutz, "Mode-locked rotating detonation waves: Experiments and a model equation," Phys. Rev. E **101**, 013106 (2020).

[37] P. Wolanski, "Rotating detonation wave stability," in *Proceedings of the 23rd International Colloquium on the Dynamics of Explosions and Reactive Systems* (2011).

[38] J. W. Bennewitz, B. R. Bigler, J. J. Pilgram, and W. A. Hargus, "Modal transitions in rotating detonation rocket engines," Int. J. Energetic Mater. Chem. Propul. **18**, 91–109 (2019).

[39] J. Koch and J. N. Kutz, "Modeling thermodynamic trends of rotating detonation engines," Phys. Fluids **32**, 126102 (2020).

[40] J. Kelly, J. Bettencourt, M. J. Johnson, and D. K. Duvenaud, "Learning differential equations that are easy to solve," in *Advances in Neural Information Processing Systems*, edited by H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (Curran Associates, Inc., 2020), Vol. 33, pp. 4370–4380.

[41] A. Ghosh, H. Behl, E. Dupont, P. Torr, and V. Namboodiri, "Steer: Simple temporal regularization for neural ODE," in *Advances in Neural Information Processing Systems*, edited by H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (Curran Associates, Inc., 2020), Vol. 33, pp. 14831–14843.

[42] C. Finlay, J.-H. Jacobsen, L. Nurbekyan, and A. Oberman, "How to train your neural ODE: The world of Jacobian and kinetic regularization," in *Proceedings of the 37th International Conference on Machine Learning, Proceedings of Machine Learning Research*, edited by H. D. III and A. Singh (PMLR, 2020), Vol. 119, pp. 3154–3164.

[43] R. Bilger, "Reaction rates in diffusion flames," Combust. Flame **30**, 277–284 (1977).

[44] P. Clavin and F. A. Williams, "Theory of premixed-flame propagation in large-scale turbulence," J. Fluid Mech. **90**, 589–604 (1979).

[45] R. Linn, "A transport model for prediction of wildfire behavior," Technical Report, Los Alamos National Laboratory, 1997.