PNAS NEXUS

PNAS Nexus, 2025, 4, pgaf018

https://doi.org/10.1093/pnasnexus/pgaf018
Advance access publication 21 January 2025
Research Report

From disorganized data to emergent dynamic models: Questionnaires to partial differential equations

David W. Sroczynski [a,b], Felix P. Kemeth [b], Anastasia S. Georgiou [b], Ronald R. Coifman [c] and Ioannis G. Kevrekidis [b,d,*]

[a]Department of Chemical and Biological Engineering, Princeton University, Princeton, NJ 08544, USA
[b]Department of Chemical and Biomolecular Engineering, Johns Hopkins University, Baltimore, MD 21218, USA
[c]Department of Mathematics, Yale University, New Haven, CT 06520, USA
[d]Department of Applied Mathematics and Statistics, Johns Hopkins University, Baltimore, MD 21218, USA
*To whom correspondence should be addressed: Email: yannisk@jhu.edu
Edited By Doraiswami Ramkrishna

Abstract

Starting with sets of disorganized observations of spatially varying and temporally evolving systems, obtained at different (also disorganized) sets of parameters, we demonstrate the data-driven derivation of parameter dependent, evolutionary partial differential equation (PDE) models capable of generating the data. This *tensor* type of data is reminiscent of shuffled (multidimensional) puzzle tiles. The *independent variables* for the evolution equations (their "space" and "time") as well as their effective parameters are all *emergent*, i.e. determined in a data-driven way from our disorganized observations of behavior in them. We use a diffusion map based *questionnaire* approach to build a smooth parametrization of our emergent space/time/parameter space for the data. This approach iteratively processes the data by successively observing them on the "space," the "time" and the "parameter" axes of a tensor. Once the data become organized, we use machine learning (here, neural networks) to approximate the operators governing the evolution equations in this emergent space. Our illustrative examples are based (i) on a simple advection–diffusion model; (ii) on a previously developed vertex-plus-signaling model of *Drosophila* embryonic development; and (iii) on two complex dynamic network models (one neuronal and one coupled oscillator model) for which no obvious smooth embedding geometry is known a priori. This allows us to discuss features of the process like symmetry breaking, translational invariance, and autonomousness of the emergent PDE model, as well as its interpretability.

Keywords: machine learning, generative models, partial differential equations, latent spaces, questionnaires

Significance Statement

The extraction of evolutionary partial differential equation (PDE) models from data is predicated on knowing the right *independent* variables: space and time coordinates. The framework of this article extends recent developments in machine-learning assisted extraction of PDEs from data to cases where the true time, space, and parameter value instances at which data were obtained are disorganized, and thus unknown. We only know labels for the spatial, temporal and parameter measurement channels. Using a Questionnaire metric, the scrambled tensor measurements are organized and smoothly embedded in an intrinsic, *emergent* physical space/time/parameter space, so that predictive dynamic models for the data (here, in the form of parameter-dependent PDEs) can be learned.

Introduction

Data science and machine learning (ML) daily expand the set of data-driven tools in the mathematical modeler's toolkit. This toolkit enables, among other tasks, the extraction of data-driven dynamic models capable of predicting the evolution of a system's response as a function of initial/boundary conditions and (possibly) external parameters. The input to this process is a (rich enough) set of time series (or image series, movies) of experimental observations of the system we wish to model.

A simple illustration is seen in Fig. 1A: a space–time plot depicting the evolution of a concentration field in a plug flow tube, governed by a scalar, 1D advection–diffusion partial differential equation (PDE). Initially, there is no tracer anywhere in the tube; then, a step change in tracer concentration is introduced at the inlet.

Given this "movie" in the form of the space–time field $u(x, t)$ we can obtain (at every x and every t) a set of measurements: u, u_t, u_{tt}, u_x, u_{xx}, u_{xxx}, etc. If we have some reason to believe that the dynamics can be modeled in the form of a PDE of the form $u_t = \mathcal{L}(u_x, u_{xx})$, then each point of the movie gives us a point in the u_t, u_x, u_{xx} space. It is clear that, with these data, the operator $\mathcal{L}$ can be approximated (fitted) as a function of "just" u_x and u_{xx}, and any off-the-shelf neural network or Gaussian Process Regression software can be used to "learn the right-hand side of

Competing Interest: The authors declare no competing interests.
Received: June 25, 2024. **Accepted:** January 7, 2025

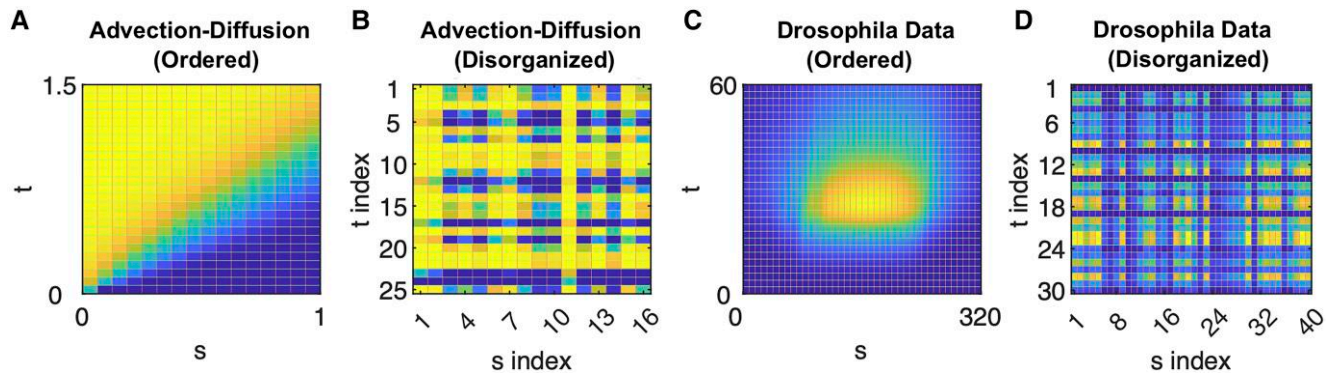

Fig. 1. An illustrative space–time caricature. A) Tracer concentration evolution during advection–diffusion in a pipe, following a step change in tracer concentration at the entrance of the pipe (see Supplementary material 1 for nondimensionalized equation, initial conditions, and boundary conditions). B) The same concentration data disorganized (shuffled) in space and in time. C) A space–time plot for chemical signal intensity data in a *Drosophila* embryonic development model (see Section 2 and Supplementary material 3). D) The same *Drosophila* data disorganized in space and time.

the PDE." Cautionary notes abound: notice, for example, how much we have already assumed, even without a formula: that it is *a first-order* PDE in time; that it is translationally invariant (its law does not explicitly depend on x) and *autonomous* (its law does not explicitly depend on t); that *the right-hand side operator only depends on* u *and its second-order spatial derivatives*; that no other variable is necessary to predict the evolution of *u*; that noise can be ignored…Yet the fact remains: in this data-driven sense, operators can be approximated, and ODEs and PDEs can be "learned" from data through, say, neural networks; this has been known and practiced for decades (2–4) (and is experiencing an explosive rebirth in the current literature (5–11)). These learned models do not need to be completely "black box" agnostic: physical knowledge can be included in "hardwiring" parts of the operator that are assumed accurately known, or learning the "calibration" of parts of the operator that are assumed only partially/qualitatively known. The term "gray box identification" is used for such algorithms (12–14). The methods to "learn" PDEs also do not need to be neural network based (15, 16).

Figure 1C is a qualitatively similar computational space–time movie: it arises in modeling the evolution in time of a chemical signal from a vertex-plus-chemistry model of *Drosophila* egg evolution (proposed in Ref. (17)). It only records a particular observation of the evolution (along with a portion of the 1D "backbone" of the equatorial slice of the egg). Without any of the myriad details, the point is that one could try and "learn" an evolution PDE for this spatiotemporal signal from the data.

Such a data-driven model can only be guaranteed, upon successful training, to be a compact summary of the data it was trained on: it can reproduce them (it can regenerate the data in the same way that a PDE solver can produce a solution for a well-posed problem). How well it can generalize (extrapolate at other initial conditions, other boundary conditions, other parameters) or whether it can assist physical understanding is, of course, another story that only starts after the small initial success of creating a compact "generator" of the training data. We will take all this "compact data generator" technology for granted, and use it in our work here.

A first sketch of the problem we want to solve is outlined in Fig. 1B: Fig. 1A has been turned into a "shredded and shuffled" puzzle. Measurements (pixels, puzzle tiles) are obtained at N_s locations in space (s_i, $i = 1, \ldots, N_s$); yet the instruments are placed at random space locations, so that while the ith instrument always measures at the same spatial location, we do not know where this location is in physical space. Figure 2 shows a plausible caricature of how such shuffled-in-space measurements may arise: consider a pipe (e.g. a reservoir) hidden under the surface of a table (e.g. a plain in Texas). We drill measurement wells on a regular grid on the plane, and number them based on the 2D plane geometry (1–16, Fig. 2B); yet the order of this numbering does not correspond to the true flow pattern under the surface, whose 1D geometry—parametrized by the pipe arclength, from I to XVI—is invisible to us. Time-series measurements of concentration (using the advection–diffusion equation (18)) ordered 1–16 with the observation geometry (Fig. 2C) gives us a violently varying (large Dirichlet energy) surface; yet if we reorder the labeling of these time series ("unshuffle" them) using space–time smoothness as our principle, we obtain a smooth surface (Fig. 2D) which visually coincides with the true space–time solution along the pipe length (Fig. 2E). Figure 2F compactly summarizes this geometry, while Fig. 2G shows "the same problem" in following flow down a human intestine: ordering sensor measurements based on physical, 3D space geometry is not faithful to the twists and turns of transport along the path of our effectively 1D digestive tract. Figure 2H shows 16 cells (after four divisions of an initial embryonic cell (1)) along with their (schematic) connectivity pattern. The 2D space we plot them in is not "the right geometry" in which to study the exchange of signals between them; the right geometry for this is the graph idealized in Fig. 2H.

In the same spirit, the N_s instruments are triggered to record at the same N_t instances in time (t_j, $j = 1, \ldots, N_t$); yet the labels j of the temporal measurements are also random (not sequential with true physical time); so, all N_s spatial channels report at time instance j, but we do not know at what actual physical time these simultaneous measurements were taken. We thus have a list of "spatial channels" and a list of "temporal channels" that index our observational tiles (and in what follows, we will make the puzzle 3D: we will add "parameter channels," performing N_p different experiments, also with disorganized labels $k = 1, N_p$; this will turn our "tiles" into "voxels").

So, we know what measurements were obtained simultaneously in time (from their index j); which come from the same experiment (from their index k); and which come from the same spatial location (from their index i); but we do not know what the actual physical time corresponds to the index j, which particular space location corresponds to the index i, and for what parameter values the measurements at parameter index k were obtained; our tensor data are "triply disorganized"—what one might call *a multipuzzle*.[a]

We want to combine (i) organizing/reconstructing the (multi)-puzzles (finding a good way to embed our measurement locations and temporal instances in an "emergent space–time" domain, or an "emergent space–time–parameter" domain) with (ii) learning

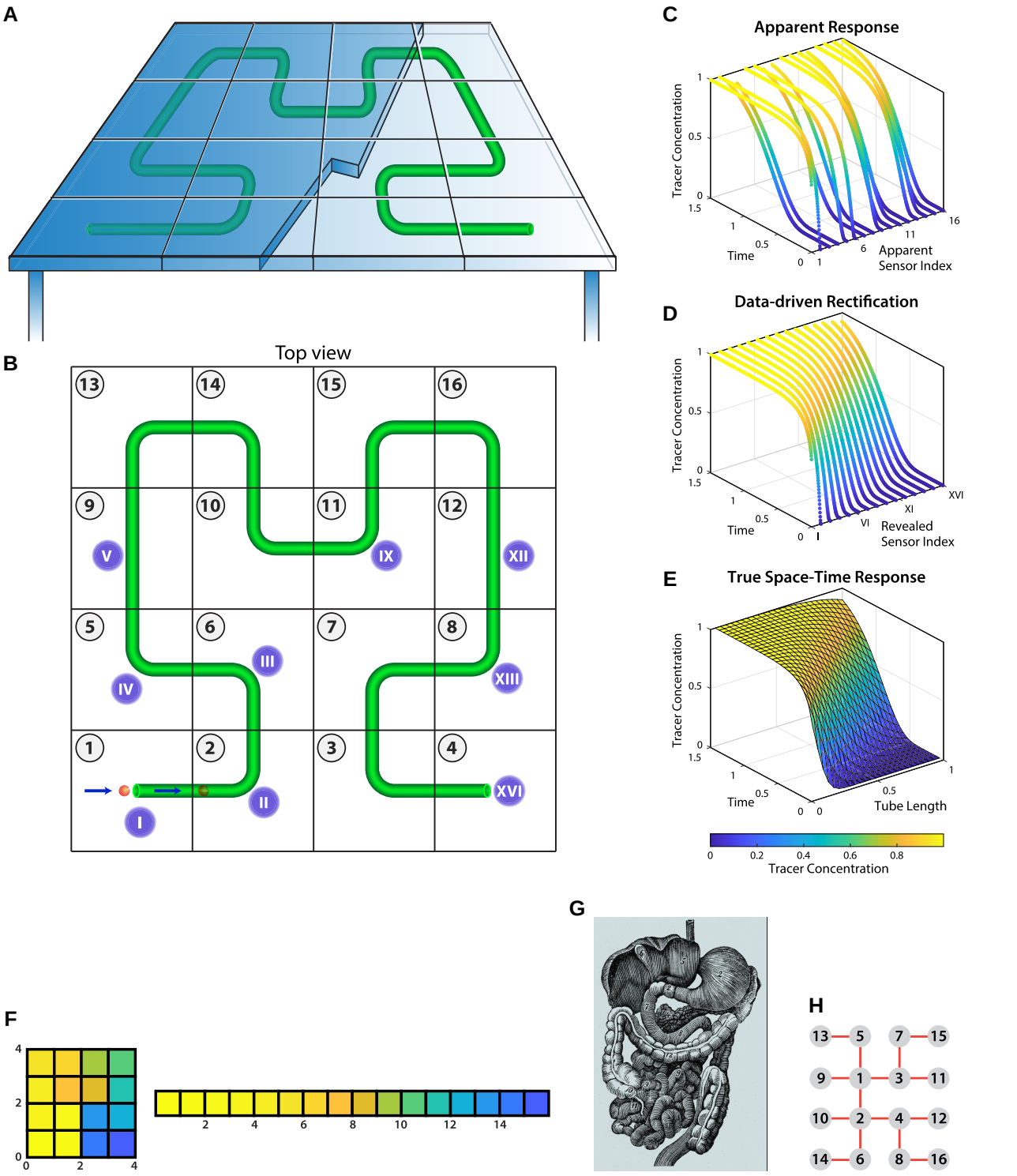

Fig. 2. A) A table covering a 1D serpentine pipe with liquid flow. B) Measurements are taken at 16 points in a regular grid, at the center of each square. Locations are initially indexed by the number in the upper left corner of the square but would be more usefully indexed by the Roman numerals following the pipe centerline. At time 0, we introduce a step change in the concentration of tracer particles at the pipe inlet. C) For our first (table-geometry) ordering, the data appears scrambled and cannot be readily interpolated. D) After data-driven rectification, the results are smooth and can be interpolated. E) Interpolation can produce the full space–time response. See Supplementary material 1 for additional details on this advection–diffusion example. F) Each square is colored by its measurement value at time $t = 0$ for both the original grid layout and the "data-revealed" sensor index. G) The curved 1D pipe example is qualitatively similar to revealing the geometry of human intestines, for example. H) Schematic caricature for a cell lineage tree in *Drosophila* germline (1).

a predictive model (an evolution equation) *in this reconstructed, data-driven, "emergent space–time"* with the ML-assisted techniques mentioned above.

While the literature of reconstructing dynamical systems from data (with known space-times!) is quite rich and fast growing (19, 20), there is also extensive literature for the mathematics and

algorithms of (instantaneous!) puzzle reconstruction (21, 22). Our algorithm of choice here for "solving the puzzle" (i.e. for the reconstruction of a useful spatial–temporal–parameter embedding from observational "puzzle voxels") will be the tensor decomposition "Questionnaires" algorithm of Ref. (23). This is illustrated, and mathematically summarized, in Supplementary Information 2; it was proposed in Ref. (23) and we have used it in the past to learn normal forms from dynamical system observations in Refs. (24, 25) (see also Refs. (26, 27)). Our algorithm for subsequently learning a predictive model for the organized data—an evolution equation in the "emergent space–time–parameter" domain—will be deep neural networks (as proposed originally in Refs. (4, 12, 28–31) and used extensively recently, e.g. in Refs. (32–35)).

The article is organized as follows: We will start with a very brief description of Diffusion Maps, and their use as part of the Questionnaires algorithm (detailed and illustrated in Supplementary material 2). We will then briefly introduce the data we will use, which come from a previously proposed/studied vertex-plus-signaling model of *Drosophila* embryonic development, described in detail in Supplementary material 3. Finally, we will describe and illustrate "learning the PDE" in the emergent domain. When a problem is even mildly nontrivial, interesting twists arise in treating it; here, these twists include (i) slight breaking of a left/right symmetry and (ii) the fact that we know that the problem is not spatially translationally invariant: it includes a physically motivated, spatially localized, temporally varying forcing term (the source of the signaling). How these two "twists" arise and are dealt with in our data-based scheme is, we believe, of some interest.

We conclude with a few of the (myriad) caveats and shortcomings of the approach. Even with those, we argue that the combination of puzzle-solving with nonlinear distributed system identification, and the ability to create "intelligent" emergent domains in which to learn smooth models, is an important pursuit, extending the tools of modern data-driven modeling. We emphasize that it is the integration of the two techniques (unsupervised and supervised learning) that we hope to demonstrate with this work, rather than the details or the exact choice of the individual components (whose exact formulation can be replaced with what is appropriate for the problem at hand).

A final note before starting: why scramble a space–time you already know? The answer is that we first validate the approach on problems where we know the solution, before it can be applied to data with hidden space-times (think of segments of broken fossils in different earth strata at different locations, or measurements more easily labeled by device name—e.g. sample point A301—rather than space location) as well as data where no obvious physical space–time exists (e.g. dynamics of networks, power networks, or physical neural networks), but which can be usefully visualized in data-driven space-times (e.g. Refs. (36, 37)). We present two additional examples in Supplementary material 5, both illustrating systems lacking an obvious spatial dimension a priori, but where an effective spatial parameterization emerges through this approach. These examples include a model of the pre-Bötzinger complex, a neuronal network in the brainstem, and the Stuart–Landau oscillators, an agent-based system of coupled ODEs. In both cases, a meaningful spatial parameterization is uncovered using the questionnaire metric. In this work, we intentionally scramble the *known* space and time data in our *Drosophila* model, not only to validate our approach but also to demonstrate its capability to reconstruct meaningful patterns from disordered, high-dimensional tensors.

Methods for data analysis

This section briefly introduces the data organization tool: the questionnaire informed metric that iteratively synthesizes and organizes information viewed along different axes of the data tensor. Since it builds upon the *Diffusion Maps* manifold learning technique, we first provide a brief overview of diffusion maps.

Manifold learning: diffusion maps

The goal of manifold learning is to discover underlying lower-dimensional intrinsic nonlinear structure in high-dimensional data. Diffusion maps (38, 39) accomplishes this by constructing a discrete approximation of the Laplacian operator on the data. When the data are sampled from a low-dimensional manifold, the discrete operator converges to the continuous Laplace–Beltrami operator *on the manifold* in the limit of infinite sample points. The discrete operator is constructed by defining a weighted graph on some N sampled data points, where the weight between points i and j is given by

$$w_{i,j} = \exp\left(-\frac{d(\mathbf{y}_i, \mathbf{y}_j)^2}{\epsilon^2}\right); \tag{1}$$

$d(\bullet, \bullet)$ represents a chosen distance metric (e.g. Euclidean in the ambient space) and ϵ represents a distance scale below which samples are considered similar. A weight of 1 indicates that two samples are identical, while a weight close to 0 indicates that two samples are very dissimilar. After some normalization, the leading nonharmonic eigenvectors $\{\phi_k\}$ of the kernel matrix, weighted by the corresponding eigenvalues $\{\lambda_k\}$, provide a new coordinate system for embedding/describing the data. Distances in this coordinate system are referred to as *diffusion distances*. Eigenvectors which do not contribute to this distance (due to low eigenvalues) can be truncated, and the reduced set of eigenvectors can serve as a proxy for the intrinsic manifold coordinates. More details can be found in Refs. (38, 39). Recent years have brought significant advancements in diffusion maps, including theoretical developments (40, 41) and practical advancements (42, 43). One such advancement, the *questionnaire-informed metric*, is the focus of the following section.

Iteratively informed geometry and the questionnaire metric

One of the key choices in the implementation of diffusion maps is that of the metric used to compare data points. In many cases, the standard Euclidean norm in ambient space is sufficient, but in certain applications (such as for very noisy or sparse datasets, or scrambled datasets where correlations between dimensions can be exploited like in our *Drosophila* data (23)) other metrics may be warranted. It may be easiest to describe this in reference to the colorful caricature in Fig. S1. Consider the case where we want to find a jointly smooth embedding for the rose color channels as well as for the blooming stage (age) channels at which we collect data. The questionnaire metric (see Supplementary material 2 and Refs. (24, 25, 36)) uses (i) the data-driven geometry of the blooming stage indices to inform the distances between the color channel indices; as well as (ii) the data-driven, now "slightly informed about blooming stage" geometry of the color channel indices to inform, in turn, the distances between the blooming stage indices. The procedure iteratively improves the joint metric until convergence. If a third "viewpoint" (in addition to color and blooming stage—e.g. possibly type of fertilizer used) is included, we iterate between all three viewpoints.

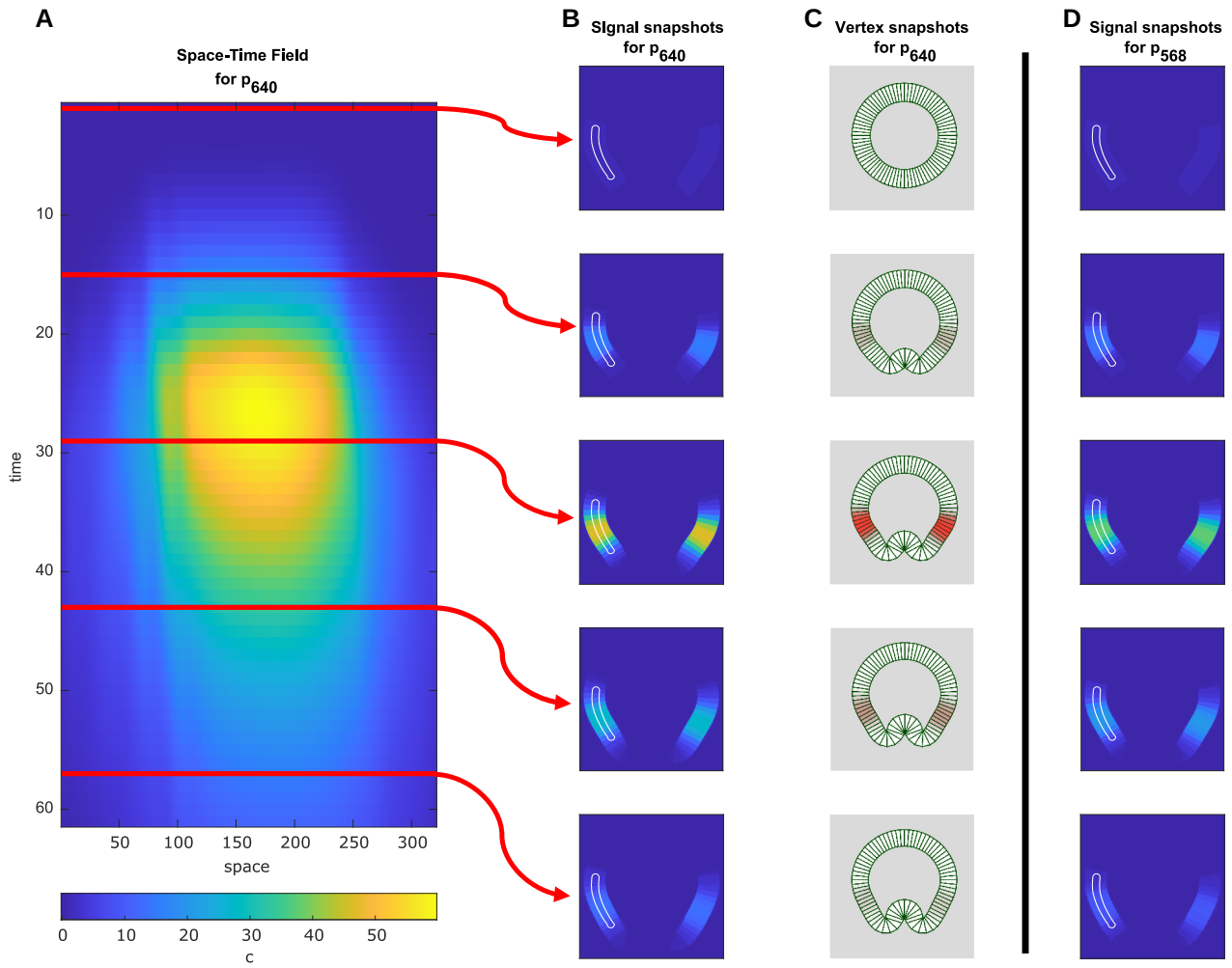

Fig. 3. A) A space–time *Drosophila* embryo chemical signal field for a particular parameter setting ($d = 0.1750$, $D_e = 0.0650$, $t_s = 40.17$). Five representative points in time are highlighted in red. B) Observation snapshots corresponding to those points in time. Data are taken from the area outlined in white. C) Corresponding snapshots, showing the location of the cell vertices. D) Signal Snapshots at the same time instances but for a different parameter setting ($d = 0.1830$, $D_e = 0.0848$, $t_s = 39.91$).

Illustrative example

Model data

To illustrate our approach, we will use data generated by a vertex-plus-signaling model of a tubular epithelium which approximates ventral furrow formation in early *Drosophila* embryos; the model was first developed in Ref. (17) (see also Refs. (44–46)). The approach combines:

1. A 2D mechanical model consisting of a ring of 80 quadrilateral cells. Since each pair of neighboring cells shares two vertices, the state space of the mechanical model is described by the positions of 160 vertices. The vertices are acted on by line tension on the edges, a stiff outer membrane, and energy penalties for deviation in the volume of each individual cell, as well as the central "yolk" they collectively envelop.
2. A chemical signal model for a protein involved in embryonic development. The (temporally varying) chemical intensity of the signal is assumed to be spatially uniform within each cell (the cells are "well mixed"). There is a source for the signal in certain cells, whose intensity is time-dependent: it grows to a maximum value at the "stopping time" t_s before decaying exponentially with first-order rate constant d. Transport between the cells is proportional to the concentration difference between them, and is characterized by an "effective diffusivity" D_e.

The overall model overlays the mechanical and chemical components to generate "videos" (series of snapshots) in the spirit of experimentally tracking the staining for the relevant protein. For more details, see Supplementary material 3. The model contains a number of constitutive parameters; here, we will fix several of them, and focus on **three** that we will allow to vary: the effective diffusivity D_e, the protein degradation rate constant d, and the stopping time t_s.

We generated data from this *Drosophila* embryo model for 1000 distinct parameter settings. For each configuration, D_e and d were generated from independent normal distributions with $\{\mu_{D_e} = 0.2, \sigma_{D_e} = 0.04\}$ and $\{\sigma_d = 0.075, \sigma_d = 0.005\}$. Settings beyond two standard deviations from the mean were discarded and redrawn. The stopping time t_s was then taken to be a prescribed function of D_e and d; the point of this is to illustrate that, *even though three parameters are varying, there is only a two-parameter family of variations*—so that our parameter settings lie on a 2D

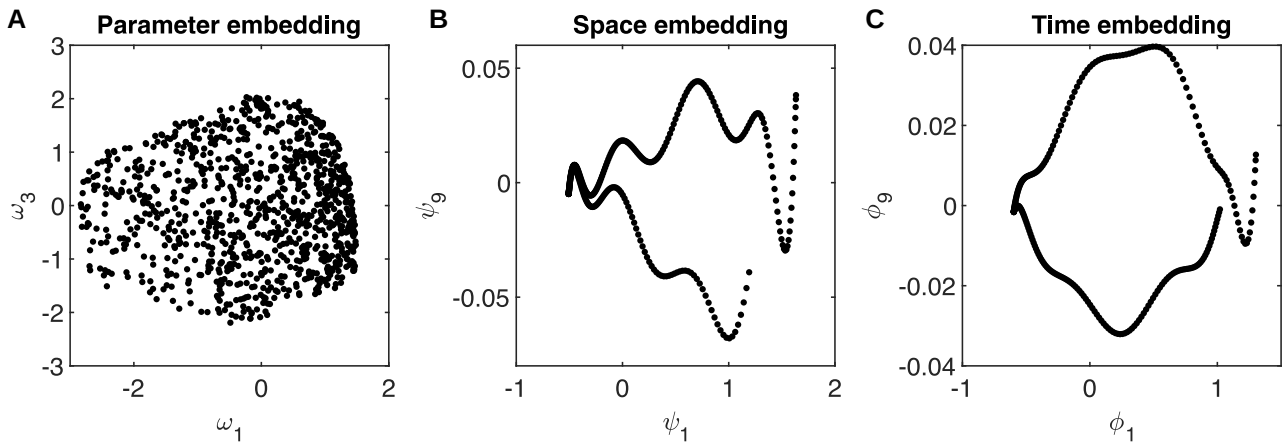

Fig. 4. The recovered questionnaire embeddings for the A) parameters, B) space, and C) time (see text).

manifold embedded in the 3D parameter space (Fig. 5A). We will thus expect our data-driven approach to recognize that the effective parameter variation is 2D.

A representative simulation output is summarized in Fig. 3C. From this type of output, we generated "videos" of just the chemical signal expression (Fig. 3B), with no specific morphology information, since the morphology evolves in the same way in all our trials. In order to simplify analysis, we sample data from a 1D "backbone curve," guided by the midpoints of the cell interface edges (see white outlines in Fig. 3B/D).

We approximate the data, sampled on $N = 360$ points along this curve of cell midpoints, using bivariate splines. With $T = 61$ snapshots for each movie, this results in a $1{,}000 \times 61 \times 360$ data tensor; each snapshot contains $N = 360$ spatial grid points. Figure 3 shows how a space–time field for a particular parameter setting corresponds to snapshots from the original video; to illustrate variability, some snapshots from a different parameter setting are also included in the last column. We then artificially scramble our *Drosophila* dataset.

Embedding results

After applying diffusion maps with *the questionnaire-informed metric* to our *scrambled Drosophila* data, a subset of the diffusion map eigenvectors provide an embedding for each axis of our data tensor (parameters, space, and time), which are shown in Fig. 4. While these embeddings may not be intuitive at first glance, they can (in this example) be rationally interpreted in terms of the underlying system.

We begin with the embedding of the parameters, since it is the simplest. Since our parameter settings were sampled from a 2D manifold, we would expect the algorithm to embed the data with only two unique coordinates. In general, taking diffusion map eigenvectors with the highest eigenvalue may not give the most parsimonious embedding, since "higher harmonics" of significant coordinates may appear before unique coordinates. Methods exist, however, to filter out such unnecessary coordinates (47). In this case, the first and third diffusion map eigenvectors were the only relevant coordinates, with eigenvector 2 being a function ("higher harmonic") of eigenvector 1. Figure 5 shows that the two recovered coordinates are bi-Lipschitz with the true parameters, with the first coordinate being mostly a function of d and the other mostly a function of D_e. Thus, with no prior assumptions on the nature of the system, we have established the effective two-dimensionality of the parameter variations, and we have revealed the underlying intrinsic organization geometry of the data in parameter space.

Given that the data are sampled from a 1D curve in space, it is reasonable to expect a 1D embedding for space; yet in this case the algorithm gives a 2D space embedding (Fig. 4B) which is locally 1D. Note the ridged "hairpin"-like shape of this embedding, which can be explained as follows: For a perfectly circular domain, reaction–diffusion dynamics on both sides of a source cell would give left–right symmetric concentration profiles. However, the asymmetric (and moving!) source cell locations introduce a symmetry breaking into the system, making the branches "above" and "below" the source close *but distinguishable*. This becomes clear in the space embedding (Fig. 6A), where ψ_1 encodes the distance to the source term and ψ_9 the induced symmetry breaking.

We therefore use the arclength along this "hairpin" to parametrize the emergent spatial geometry of the data. An effective coordinate $\tilde{\psi}$ is extracted using diffusion maps on the curve in Fig. 6A with a nearest neighbor similarity measure, and shown in Fig. 6B as a function of the arclength s along the cell centerline from which data was taken—notice the one-to-one correspondence. We use this coordinate $\tilde{\psi}$ as the emergent, data-driven space coordinate (35) in which to learn the dynamics of $c(\tilde{\psi}, t)$ (Fig. 6C).

The embedding of the time samples (Fig. 4C) also has a similar quirk, in that it requires two dimensions. The first embedding coordinate roughly follows the overall chemical concentration, which starts at zero, rises to a maximum near t_s, and then fades back asymptotically to zero. Because the numerical experiment is stopped at finite time, the embedding coordinate never reaches its original value, but comes close. There is a need for a second embedding coordinate (ϕ_9) which captures the fact that the spatial distribution of chemical concentrations is different when the overall level is rising (more tightly focused around the source cells) than when it is falling (more spread out due to diffusion between cells). In this case, the diffusion is not that strong, so the difference is slight, which is why this second coordinate "shows up" later on in the spectral hierarchy as ϕ_9. Essentially, the time evolution of this system has been characterized by the algorithm as a "skinny" loop.

Similarly to our approach for the space embedding, we can extract a coordinate $\tilde{\phi}$ from this skinny loop, to obtain a 1D embedding for emergent time (Fig. 7B). We also show the data in this final *emergent space–time* in Fig. 7C. We emphasize that these embeddings are the output of diffusion maps with the

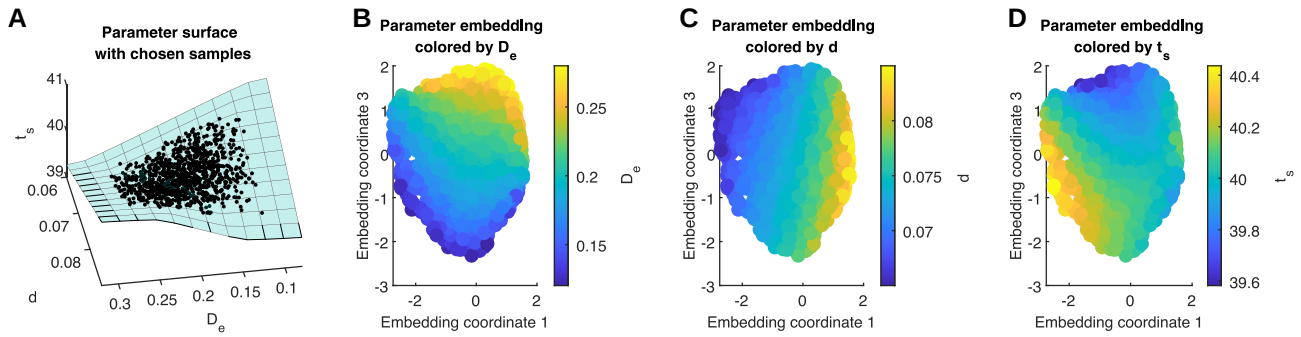

Fig. 5. A) Parameter settings of the observations (black dots) in D_e,d,t_s-space, with the blue surface showing the 2D manifold those parameters were drawn from. B–D) Data-driven parameter embedding coordinates (as in Fig. 4A) colored by D_e, d, and t_s. The embedding is one-to-one with the 2D parameter domain surface.

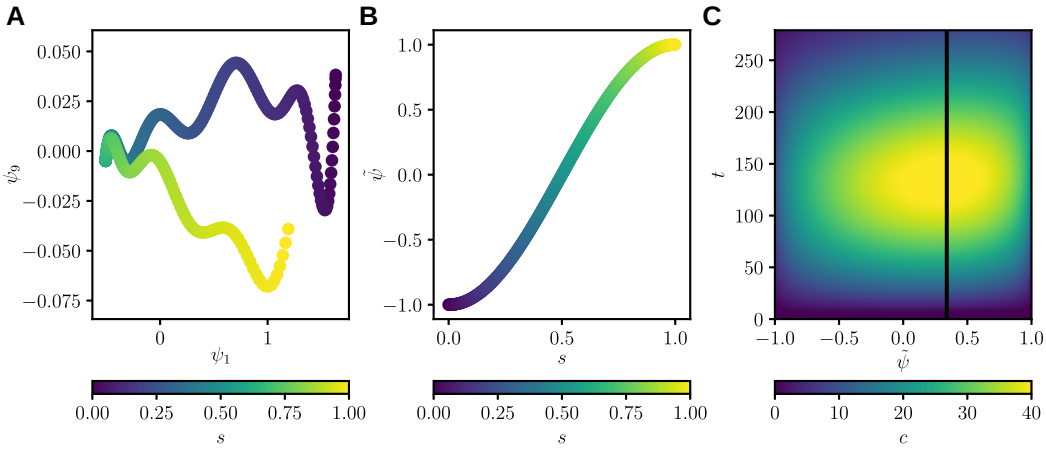

Fig. 6. A: Data-driven space embedding (as in Fig. 4B) obtained from the questionnaires, colored with the position s along the cell section centerline from which the data were taken. B: Arc length $\tilde{\psi}$ extracted from the space embedding ψ, as a function of the position s. C: Concentration data parametrized by the extracted embedding arclength $\tilde{\psi}$. The black vertical line corresponds to the position of the minimum of ψ_1; that is, to the left edge of the embedding shown on the left.

questionnaire-informed metric, and it is these embeddings that reorganize our data tensor to appear smooth before we can learn the underlying dynamics.

Learning the dynamics in the emergent coordinates

We start the section by approximating the evolution operator in emergent space but still in physical time; we pose a distributed parameter model (a PDE) whose right-hand side is approximated by a fully connected feed forward neural network (in emergent space) (3, 19). We approximate the dynamics of the concentration for a single parameter setting through a PDE

$$\frac{\partial c}{\partial t} = f\left(c, \frac{\partial c}{\partial \tilde{\psi}}, \ldots, \frac{\partial^n c}{\partial \tilde{\psi}^n}\right), \qquad (2)$$

where f is represented by a neural network and $c = c(\tilde{\psi}, t)$. Here, we use $n = 3$ derivatives, estimated using finite differences. We therefore resample the data on an equally spaced grid in $\tilde{\psi}$ using a bivariate spline approximation, and also on $T = 1{,}500$ equally spaced points in actual time.

f is composed of three fully connected hidden layers with 966 neurons each, with one output layer containing a single node. Each hidden layer is followed by a Swish activation function (48). The model is optimized using the PyTorch framework (49) and a Adam (50) optimizer with the default hyperparameters, based on the mean squared error of the predicted and true $\frac{\partial c}{\partial t}$. The (actual) time derivatives are estimated using finite differences in (actual) time. The initial learning rate is set to 0.005, and subsequently halved when the training loss did not decrease for 75 epochs. The model is trained for a total of 750 epochs using a batch size of 256. Overfitting was assessed using a held out validation set composed of all spatial points of the respective last 300 snapshots (20% of the data). Note that *we do not provide a chemical source term* in the model. We therefore ignore a narrow space–time corridor surrounding the source location, and learn the (translationally invariant) PDE in its complement. Furthermore, boundary conditions are not in principle available for the learned model. We therefore provide, in lieu of boundary conditions, narrow "boundary corridors" informed by the data. As in Ref. (35), we regularize the outputs of the learned model using a truncated singular value decomposition. Finally, we integrate an initial c profile using the learned model.

Note that here we learned the dynamics, cf. Eq. 2, at just a single parameter setting, and took the temporal ordering of the snapshots as *known and given*. However, if the true times are not known, we can instead construct the model to integrate in emergent time. As discussed above, scrambled temporal data result in a hairpin embedding ϕ similar to the space embedding, so we use a similar emergent coordinate $\tau = \tilde{\phi}$ (see Fig. 7), and learn a PDE operator (a "right-hand side") in this emergent time.

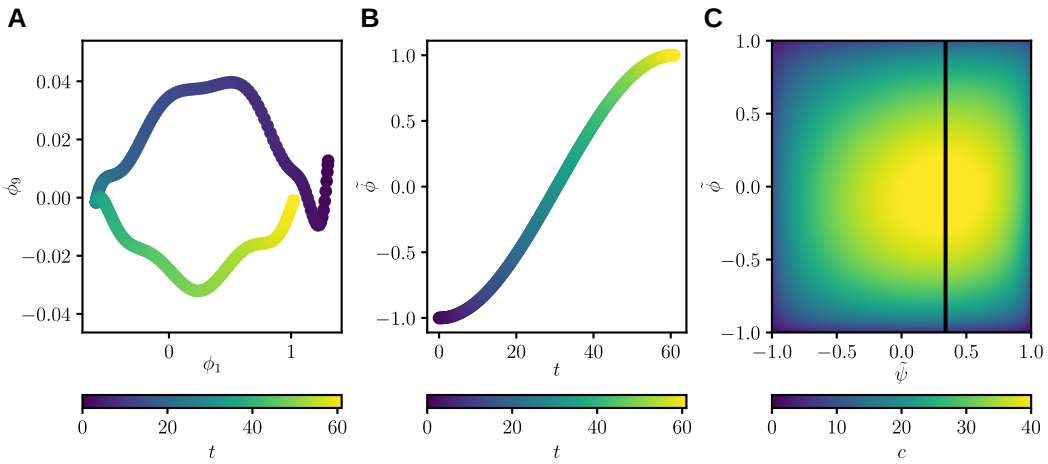

Fig. 7. A: Data-driven time embedding (as in Fig. 4C) obtained from the questionnaires, colored with the true time t. B: Embedding arclength $\tilde{\phi}$ extracted from the time embedding ϕ, as a function of the true time t. C: Concentration data parametrized by the extracted arclengths $\tilde{\psi}$ and $\tilde{\phi}$. The black vertical line corresponds to the position of the minimum of ψ_1 as above.

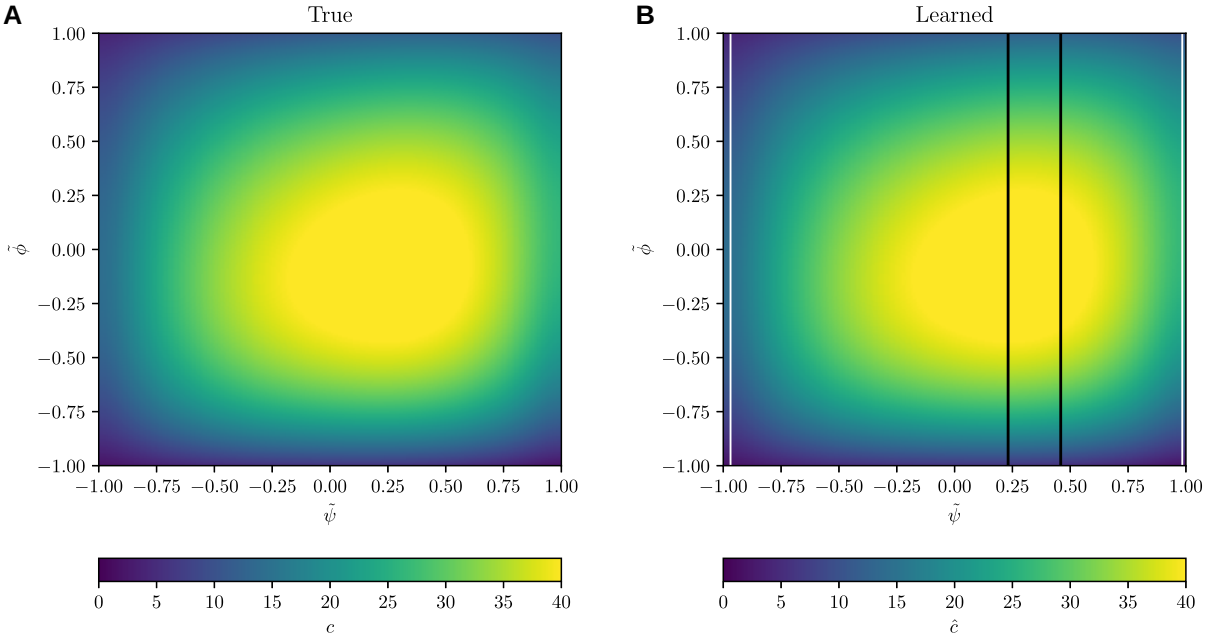

Fig. 8. A: True data as shown in Fig. 7C. B: Simulation results using the nonautonomous ML-learned emergent PDE model, Eq. 17, using the same initial snapshot. The boundary conditions were provided in the area along the left and right edges as defined by the vertical white lines. Data for the PDE source term in the area between the two vertical black lines (bracketing the time-dependent source) were also provided to account the nonautonomous nature of the PDE. Note the visual agreement with the true data.

$$\frac{\partial c}{\partial \tilde{\phi}} = f\left(c, \frac{\partial c}{\partial \tilde{\psi}}, \ldots, \frac{\partial^n c}{\partial \tilde{\psi}^n}\right) \tag{3}$$

with $c = c(\tilde{\psi}, \tau)$. The results of integrating this model are shown in Fig. 8. Data-informed boundary corridors, as well as the source term are provided.

If the source term is not given, the predictions of the learned (translationally invariant) PDE simulated over the entire spatial domain are simply wrong. This becomes obvious when using the learned model to predict the dynamics over the entire domain (only providing boundary conditions at the edges, but *not the source information*). Alternatively, one can extend this approach *to learn the source term in the corridor* I_s as well; for further results and discussion of learning a nonautonomous dynamical system, see Supplementary Information 4.

Future work will focus on incorporating the emergent parameter coordinates, $\boldsymbol{\pi}$, into the learning process,

$$\frac{\partial c}{\partial \tau} = f\left(c, \frac{\partial c}{\partial \tilde{\psi}}, \ldots, \frac{\partial^n c}{\partial \tilde{\psi}^n}; \boldsymbol{\pi}\right), \tag{4}$$

providing a fully data-driven system identification framework.

The hidden geometry of a complex neuronal network

While we artificially scrambled the known space–time of the *Drosophila* data, there are systems where no clear spatial geometry

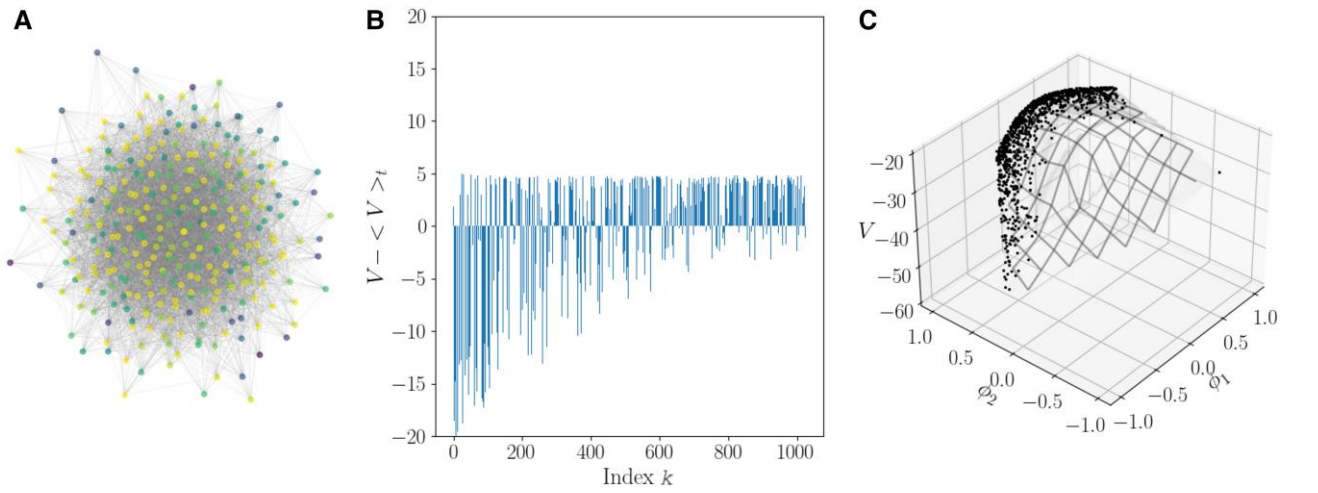

Fig. 9. A: Network representation of (a subsample of) the pre-Bötzinger complex, with nodes colored by their potential at a particular phase of their collective synchronized oscillation. The snapshot, visualized as a random arrangement of nodes, appears disordered and lacks smoothness, making it challenging to learn the dynamics. (For visualization purposes, only 308 out of the 1,024 nodes and 10% of the edge connections are shown in the left network representation.) B: Instantaneous neuronal voltage (minus the mean voltage at this particular oscillation phase) plotted at the same time instance as on the left, ordered by node index k. The state remains irregular, with no visible smooth spatial pattern. C: Voltage (z-axis) plotted as a function of the two emergent coordinates (discovered by the questionnaire metric) that parameterize the "emergent spatial" embedding of the 1,024 neurons. The state now appears 2D and smooth (moving up and down like a "waving flag" when viewed as a function of time); this description enables the ML-assisted learning of the underlying PDE dynamics.

is known a priori and our proposed approach becomes necessary. Networks are a prime example, as behavioral connectivity provides the natural representation of the underlying geometry, rather than proximity in some physical embedding space.

The pre-Bötzinger complex is a neuronal network in the brainstem responsible for generating and modulating respiratory rhythms in mammals (51). This network is typically modeled as a Chung–Lu type network (varying number of connections between nodes) of Hodgkin–Huxley neurons, each with a potential V_k (35, 52, 53). Here, we depict a model of the dynamics of V_k for $k = 1, 024$ neurons. When visualized as a network (using the networkx python package (54)) or plotted by neuron index (see Fig. 9A,B), the dynamics appear complex and disordered: the neuron index does not embody a meaningful spatial dimension. Yet, by applying the questionnaire metric using the ensemble of individual neuron oscillations as our data points, we uncover an "emergent space" parameterized by *two emergent spatial coordinates* (see Fig. 9C). In this space, the state of the 1,024 neurons can be embedded on a smooth surface. Over time, this surface exhibits wave-like motion, enabling the underlying dynamics to be learned effectively in the form of a PDE in this newly discovered "emergent" space. More details on this example, as well as an additional "phase plus amplitude" coupled oscillator network example, can be found in Supplementary Information 5.

Conclusions

The mathematics underpinning the data-driven solution of puzzles have started, in recent years, to provide increasingly sophisticated puzzle reconstructions, including cases of missing data. Even cases where different parts of the puzzle have been observed through different sensors (so, "puzzle fusion") are starting to appear in the literature (55).

Our purpose here was to combine a data organization technology (Questionnaires) with the (ML assisted) construction of predictive mathematical models. More specifically, these models came in the form of differential equations (here, PDEs): models which, given a few initial/boundary conditions, allow us to reconstruct the entire puzzle. In this sense, what we present can be thought of as a combination of data organization and "boosted" data compression: now, with very few data (initial/boundary conditions) and an evolution law (here, a parabolic PDE) we can reconstruct good approximations of all the missing data, and even sometimes extrapolate successfully. We should stress again that what we did would be *much easier* if we had explicit time/space information, as is the case in Refs. (6, 56, 57); here we had to invent the data-driven, emergent space–time in which the data appear smooth, and where, therefore, a parabolic PDE type model can be postulated. This "boosted" data compression, in the form of "very few data plus evolution law" can now be used to interpolate in parameter space or in emergent physical space–time; one may even attempt to extrapolate (up to when singularities may arise). More importantly, the approach naturally allows for "physics infusion"—if one has an informed guess of what the actual independent space variable should be, or of what an approximate closed form of the underlying dynamic model could be, this information can be included in the process in a "gray box" identification scheme (12, 13, 58, 59) that will *calibrate* the partial physical knowledge to the quantitative truth (the data) in the form of a multifidelity calibration problem.

It is important also to note how some crucial assumptions (homogeneity of the emergent space, or autonomousness, i.e. homogeneity in emergent time) shape the entire process; if there is reason to believe they do not hold, then fitting a space–time homogeneous predictive model to the data will reproduce the (training) data, interpolate between them, but fail to extrapolate/generalize. Among different equally successful interpretations (different sets of physically interpretable quantities that are also one-to-one with the data-driven observables), we can choose the one with the best numerical behavior (best condition number; best Lipschitz constants (60)). When trying to understand/rationalize the dynamics learned in emergent space, one often resorts to sparse identification: fitting the data well with "a few" common dictionary terms "ought to be the right physical interpretation." Yet it may be (at least a little) presumptuous to expect the truth to be

parsimoniously expressible in our everyday favorite dictionaries. Ultimately, the dynamic behavior does not have to appear simple *in our own favorite current language*. Following P.A.M. Dirac "…now we have to change the principle of simplicity to the principle of mathematical beauty." (61) It is then *the language in which the dynamics is beautiful* that we should strive to formulate—the transformation to the space in which the evolution is isospectral, as in the Lax Pair formulation of integrable systems, or the space in which "troubles melt like lemon drops," as Dorothy would sing in the Wizard of Oz.

Note

[a] Two (hopefully informative) visual caricatures of our "shuffling" process that results in disorganized data in two dimensions are provided in Supplementary Information Part 2 as Fig. S1 followed by Fig. S2.

Acknowledgments

The authors are grateful to Prof. S. Shvartsman and Dr. M. Misra for stimulating discussions and for graciously providing their simulation code for *Drosophila* epithelial shape transformations. They are also grateful to Dr. O. Yair and Prof. R. Talmon of the Technion for the original implementation of questionnaires used.

Supplementary Material

Supplementary material is available at *PNAS Nexus* online.

Funding

This work was partially supported by the US AFOSR (FA9550-21-1-0317, FA9550-20-1-0288) and by the DARPA Atlas program (W911NF21C0010).

Author Contributions

I.G.K. designed research. D.W.S., F.P.K., A.S.G., and I.G.K. performed research, analyzed data, and wrote the article. R.R.C. contributed analytic tools and analyzed data.

Preprints

This manuscript was posted as a preprint with DOI 10.48550/arXiv.2204.11961.

Data Availability

The data that support the findings of this study are openly available at the repository on Github at drosophila-emergentpdes.

References

1 Diegmiller R, Nunley H, Shvartsman SY, Alsous JI. 2022. Quantitative models for building and growing fated small cell networks. *Interface Focus*. 12(4):20210082. https://doi.org/10.1098/RSFS.2021.0082.
2 Krischer K, *et al.* 1993. Model identification of a spatiotemporally varying catalytic reaction. *AIChE J*. 39(1):89–98.
3 Rico-Martínez R, Krischer K, Kevrekidis IG, Kube MC, Hudson JL. 1992. Discrete- vs. continuous-time nonlinear signal processing of Cu electrodissolution data. *Chem Eng Commun*. 118(1):25–48.
4 González-García R, Rico-Martínez R, Kevrekidis IG. 1998. Identification of distributed parameter systems: a neural net based approach. *Comput Chem Eng*. 22(SUPPL.1):S965–S968.
5 Brunton SL, Noack BR, Koumoutsakos P. 2020. Machine learning for fluid mechanics. *Annu Rev Fluid Mech*. 52(1):477–508.
6 Lu L, Jin P, Pang G, Zhang Z, Karniadakis GE. 2021. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nat Mach Intell*. 3(3):218–229.
7 Karniadakis GE, *et al.* 2021. Physics-informed machine learning. *Nat Rev Phys*. 3(6):422–440.
8 Rudy SH, Brunton SL, Proctor JL, Kutz JN. 2017. Data-driven discovery of partial differential equations. *Sci Adv*. 3(4):e1602614.
9 Zhang J, Ma W. 2020. Data-driven discovery of governing equations for fluid dynamics based on molecular simulation. *J Fluid Mech*. 892:A5.
10 Raissi M. 2018. Deep hidden physics models: deep learning of nonlinear partial differential equations. *J Mach Learn Res*. 19:1–24.
11 Zhao H, Storey BD, Braatz RD, Bazant MZ. 2020. Learning the physics of pattern formation from images. *Phys Rev Lett*. 124:060201.
12 Rico-Martinez R, Anderson JS, Kevrekidis IG. Continuous-time nonlinear signal processing: a neural network based approach for gray box identification. In: *Neural Networks for Signal Processing—Proceedings of the IEEE Workshop*. IEEE, 1994. p. 596–605. https://doi.org/10.1109/nnsp.1994.366006.
13 Lovelett RJ, Avalos JL, Kevrekidis IG. 2020. Partial observations and conservation laws: gray-box modeling in biotechnology and optogenetics. *Ind Eng Chem Res*. 59(6):2611–2620.
14 Malani S, *et al.* 2023. Some of the variables, some of the parameters, some of the times, with some physics known: identification with partial information. *Comput Chem Eng*. 178:108343.
15 Long D, Mrvaljevic N, Zhe S, Hosseini B. 2024. A kernel framework for learning differential equations and their solution operators. *Phys D: Nonlinear Phenom*. 460:134095. https://doi.org/10.1016/j.physd.2024.134095.
16 Brunton SL, Proctor JL, Kutz JN. 2016. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proc Natl Acad Sci U S A*. 113(15):3932–3937.
17 Hocevar Brezavšček A, Rauzi M, Leptin M, Ziherl P. 2012. A model of epithelial invagination driven by collective mechanics of identical cells. *Biophys J*. 103(5):1069.
18 Rawlings JB. *Chemical reactor analysis and design fundamentals*. Nob Hill Publishing, 2012.
19 González-García R, Rico-Martínez R, Kevrekidis IG. 1998. Identification of distributed parameter systems: a neural net based approach. *Comput Chem Eng*. 22(nil):S965–S968.
20 Thiem TN, Kemeth FP, Bertalan T, Laing CR, Kevrekidis IG. 2021. Global and local reduced models for interacting, heterogeneous agents. *Chaos: Interdiscip J Nonlin Sci*. 31(7):073139.
21 Huroyan V, Lerman G, Wu H-T. 2020. Solving jigsaw puzzles by the graph connection Laplacian. *SIAM J Imaging Sci*. 13(4):1717–1753.
22 Sholomon D, David OE, Netanyahu NS. 2016. An automatic solver for very large jigsaw puzzles using genetic algorithms. *Genet Program Evolvable Mach*. 17(3):291–313.
23 Ankenman JI. 2014. Geometry and analysis of dual networks on questionnaires [PhD thesis]. Yale University.
24 Yair O, Talmon R, Coifman RR, Kevrekidis IG. 2017. Reconstruction of normal forms by learning informed observation geometries from data. *Proc Natl Acad Sci U S A*. 21:E7865.
25 Sroczynski DW, Yair O, Talmon R, Kevrekidis IG. 2018. Data-driven evolution equation reconstruction for parameter-dependent nonlinear dynamical systems. *Isr J Chem*. 58:787.

26 Moon KR, *et al.* 2019. Visualizing structure and transitions in high-dimensional biological data. *Nat Biotechnol*. 37(12): 1482–1492.

27 Gigante S, Charles AS, Krishnaswamy S, Mishne G. Visualizing the phate of neural networks. In: Wallach H, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox E, Garnett R, editors. *Advances in neural information processing systems*. Vol. 32. Curran Associates, Inc., 2019. https://proceedings.neurips.cc/paper/2019/file/b4d168b48157c623fbd095b4a565b5bb-Paper.pdf.

28 Hudson JL, *et al.* 1990. Nonlinear signal processing and system identification: applications to time series from electrochemical reactions. *Chem Eng Sci*. 45(8):2075–2081.

29 Rico-Martinez R, Krischer K, Kevrekidis IG, Kube MC, Hudson JL. 1992. Discrete- vs. Continuous-time nonlinear signal processing of Cu electrodissolution data. *Chem Eng Commun*. 118(1):25–48.

30 Rico-Martinez R, Kevrekidis IG. 1995. Nonlinear system identification using neural networks: dynamics and instabilities. In: Bulsari AB, editor. *Neural networks for chemical engineers*. Chapter 16. Elsevier Science B.V., 1995. p. 409–442.

31 Anderson JS, Kevrekidis IG, Rico-Martinez R. 1996. A comparison of recurrent training algorithms for time series analysis and system identification. *Comput Chem Eng*. 20:S751–S756.

32 Arbabi H, Bunder JE, Samaey G, Roberts AJ, Kevrekidis IG. 2020. Linking machine learning with multiscale numerics: data-driven discovery of homogenized equations. *JOM*. 72(12):4444–4457.

33 Arbabi H, Kevrekidis IG. 2021. Particles to partial differential equations parsimoniously. *Chaos*. 31(3):033137.

34 Lee S, Kooshkbaghi M, Spiliotis K, Siettos CI, Kevrekidis IG. 2020. Coarse-scale PDEs from fine-scale observations via machine learning. *Chaos*. 30(1):013141.

35 Kemeth FP, *et al.* 2022. Learning emergent partial differential equations in a learned emergent space. *Nat Commun*. 13(1): 3318. https://doi.org/10.1038/s41467-022-30628-6.

36 Mishne G, *et al.* 2016. Hierarchical coupled-geometry analysis for neuronal structure and activity pattern discovery. *IEEE J Sel Top Signal Process*. 10(7):1238–1253.

37 Kemeth FP, *et al.* 2018. An emergent space for distributed data with hidden internal order through manifold learning. *IEEE Access*. 6:77402–77413.

38 Coifman RR, Lafon S. 2006. Diffusion maps. *Appl Comput Harmon Anal*. 21:5.

39 Lafon S. 2004. [Ph.D. thesis]. Yale University.

40 García Trillos N, Gerlach M, Hein M, Slepčev D. 2019. Error estimates for spectral convergence of the graph Laplacian on random geometric graphs toward the Laplace–Beltrami operator. *Foundations Comput Math*. 20(4):827–887.

41 Teng Y, Sachdev S, Scheurer MS. 2023. Clustering neural quantum states via diffusion maps. *Phys Rev B*. 108(20):205152. https://doi.org/10.1103/physrevb.108.205152.

42 Pillaud-Vivien L, Bach F. 2023. Kernelized diffusion maps, arXiv, arXiv:2302.06757, https://doi.org/10.48550/arXiv.2302.06757, preprint: not peer reviewed.

43 Long AW, Ferguson AL. 2019. Landmark diffusion maps (L-dMAPS): accelerated manifold learning out-of-sample extension. *Appl Comput Harmon Anal*. 47(1):190–211.

44 Polyakov O, *et al.* 2014. Passive mechanical forces control cell-shape change during drosophila ventral furrow formation. *Biophys J*. 107(4):998.

45 Misra M, Audoly B, Kevrekidis IG, Shvartsman SY. 2016. Shape transformations of epithelial shells. *Biophys J*. 110(7):1670.

46 Misra M. 2016. Vertex models and three-dimensional epithelial morphogenesis [PhD thesis]. Princeton University.

47 Dsilva CJ, Talmon R, Coifman RR, Kevrekidis IG. 2018. Parsimonious representation of nonlinear dynamical systems through manifold learning: a chemotaxis case study. *Appl Comput Harmon Anal*. 44:759.

48 Ramachandran P, Zoph B, Le QV. 2017. Swish: a self-gated activation function, arXiv, preprint.

49 Paszke A, *et al.* 2019. PyTorch: an imperative style, high-performance deep learning library. Proceedings of the 33rd International Conference on Neural Information Processing Systems. Article No. 721. p. 8026–8037.

50 Kingma DP, Ba J. 2015. Adam: a method for stochastic gradient descent. ICLR: international conference on learning representations. p. 1–15.

51 Smith JC, Ellenberger HH, Ballanyi K, Richter DW, Feldman JL. 1991. Pre-Bötzinger complex: a brainstem region that may generate respiratory rhythm in mammals. *Science*. 254(5032):726–729.

52 Rubin J, Terman D. 2002. Synchronized activity and loss of synchrony among heterogeneous conditional oscillators. *SIAM J Appl Dyn Syst*. 1(1):146–174.

53 Laing CR, Zou Y, Smith B, Kevrekidis IG. 2012. Managing heterogeneity in the study of neural oscillator dynamics. *J Math Neurosci*. 2(1):5.

54 Hagberg AA, Schult DA, Swart PJ. Exploring network structure, dynamics, and function using NetworkX. In: Varoquaux G, Vaught T, Millman J, editors. *Proceedings of the 7th Python in Science Conference (SciPy2008)*. Pasadena, CA, USA, 2008. p. 11–15.

55 Peterfreund E, *et al.* 2023. Gappy local conformal auto-encoders for heterogeneous data fusion: in praise of rigidity, arXiv, preprint.

56 Li Z, *et al.* 2021. Fourier neural operator for parametric partial differential equations. International Conference on Learning Representations.

57 Fabiani G, Kevrekidis IG, Siettos C, Yannacopoulos AN. 2025. RandONets: Shallow networks with random projections for learning linear and nonlinear operators. *J Comput Phys*. 520: 113433. https://doi.org/10.1016/j.jcp.2024.113433.

58 Martin-Linares CP, Psarellis YM, Karapetsas G, Koronaki ED, Kevrekidis IG. 2023. Physics-agnostic and physics-infused machine learning for thin films flows: modelling, and predictions from small data. *J Fluid Mech*. 975:A41. https://doi.org/10.1017/jfm.2023.868.

59 Psarellis YM, *et al.* 2024. Data-driven discovery of chemotactic migration of bacteria via coordinate-invariant machine learning. *BMC Bioinformatics*. 25(1):337. https://doi.org/10.1186/s12859-024-05929-w.

60 Koelle SJ, Zhang H, Meila M, Chen Y-C. 2022. Manifold coordinates with physical meaning. *J Mach Learn Res*. 23(133):1–57.

61 Dirac PAM. 1939. The relation between mathematics and physics. *Proc R Soc Edinb*. 59:122–129.