Black and gray box learning of amplitude equations: Application to phase field systems

Felix P. Kemeth[1], Sergio Alonso[2], Blas Echebarria[2], Ted Moldenhawer[3], Carsten Beta[3] and Ioannis G. Kevrekidis[1,*]

[1]*Department of Chemical and Biomolecular Engineering, Whiting School of Engineering, Johns Hopkins University, 3400 North Charles Street, Baltimore, Maryland 21218, USA*

[2]*Department of Physics, Universitat Politècnica de Catalunya, Carrer de Jordi Girona 1-3, 08034 Barcelona, Spain*

[3]*Institut für Physik und Astronomie, Universität Potsdam, Karl-Liebknecht-Straße 24/25, 14476 Potsdam-Golm, Germany*

(Received 10 July 2022; accepted 26 January 2023; published 16 February 2023)

We present a data-driven approach to learning surrogate models for amplitude equations and illustrate its application to interfacial dynamics of phase field systems. In particular, we demonstrate learning effective partial differential equations describing the evolution of phase field interfaces from full phase field data. We illustrate this on a model phase field system, where analytical approximate equations for the dynamics of the phase field interface (a higher-order eikonal equation and its approximation, the Kardar-Parisi-Zhang equation) are known. For this system, we discuss data-driven approaches for the identification of equations that accurately describe the front interface dynamics. When the analytical approximate models mentioned above become inaccurate, as we move beyond the region of validity of the underlying assumptions, the data-driven equations outperform them. In these regimes, going beyond black box identification, we explore different approaches to learning data-driven corrections to the analytically approximate models, leading to effective gray box partial differential equations.

DOI: 10.1103/PhysRevE.107.025305

I. INTRODUCTION

Phase field models provide an effective mathematical framework to investigate the dynamics of interfacial boundaries [1]. They have been successfully used in fields such as material science, to describe phase separation [2] and melting processes in alloys [3–5], the formation of microstructures [6], solidification [7], solute precipitation [8], and crack propagation [9] or to model grain growth [10,11], as well as in biomechanics, for example, to model the development of fractures [12], cell migration [13–18], tumor growth [19], dendrite growth [20], and mechanotransduction [21], to name a few. See Ref. [22] for a recent review. Such phase field models typically rely on a phase ϕ, ranging from -1 to 1 (or 0 to 1). One then classifies a part of the domain where $\phi \approx -1$ as one state, whereas the part where $\phi \approx 1$ corresponds to a different state, both phases being separated by an interface, where the phase ϕ transitions sharply from one state to the other.

Enormous effort has been invested into deriving evolution equations for the position of such interfaces; a prominent example involves the reduction of the dynamics of binary alloys [23]. The same is true for the derivation of equations like the Kuramoto-Sivashinsky equation, describing the height of a thin water film flowing down an inclined surface [24,25], thereby reducing the dynamics of a free boundary problem to an amplitude equation. Envelope equations approximating the long-wavelength motions of fluids or of Fermi-Pasta-Ulam-Tsingou solutions [26,27] also fall under the class of systems, where the techniques described here are applicable. Having access to effective interface (or amplitude or modulation envelope) equations not only significantly reduces the computational cost by reducing the problem dimension, but also facilitates a more detailed investigation of parametric bifurcations of the interface.

The derivations of effective interface equations are typically tedious and impose restrictions on the parameter regime for which they are valid [23,28–31]. In this article we present a data-driven alternative for the identification of interface equations. We illustrate our approach on an Allen-Cahn phase field model [3–5], where an analytic derivation of approximate front partial differential equations (PDEs) describing the interface dynamics exists [30]. One such interface equation is the Kardar-Parisi-Zhang (KPZ) equation, a low-order approximation describing the interface dynamics in a relatively narrow parameter regime [32]. By learning the PDE in a data-driven way [33–36], we demonstrate the data-driven identification of an interface evolution equation that surpasses the accuracy of analytical interface models.

Note that there is a myriad of different approaches on how to identify the right-hand side of a PDE from data, such as sparse identification of nonlinear dynamical systems using dictionaries [37,38], PDE-net [39], DeepONet [40], Fourier neural operators [41], and physics-informed neural networks [42], among others. It is worth noting that learning operators through neural networks were proposed and implemented by Kevrekidis and co-workers [33–36]. In the context of phase field systems, the authors of Ref. [43] learn the governing PDE law for various types of phase field systems using a set of different neural network approaches. Our approach differs in that, for a phase field system example, we first reduce the dynamics to the interface and then learn the effective PDE for this lower-dimensional problem only. Here we choose to represent the right-hand side of the PDE operator (the law of the PDE) describing the interface dynamics through a feedforward neural network [33–36].

*yannisk@jhu.edu

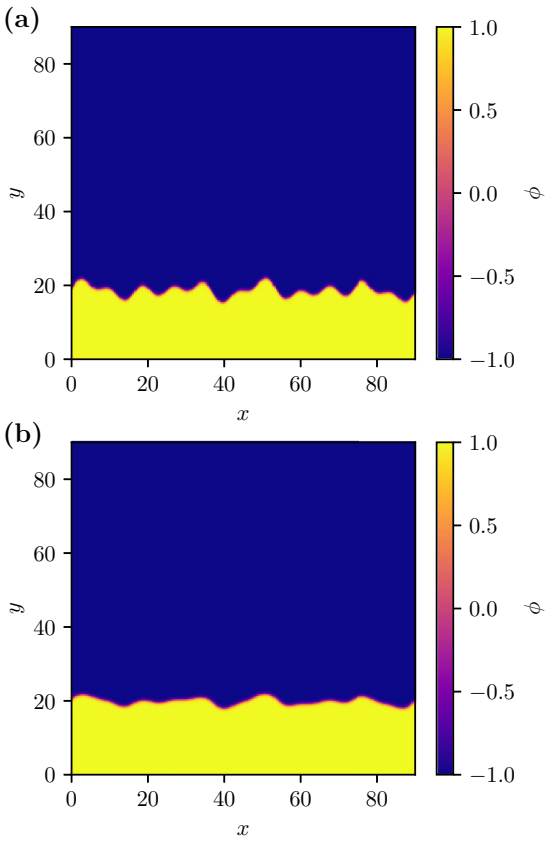

FIG. 1. (a) Initial profile of the phase field on a square domain of length $L = 90$ with a resolution of 400×400 pixels. (b) Profile after integrating the snapshot in (a) for $T = 25$ dimensionless time units using the phase field system (1) with $a = -0.1$ and $D = 0.1$. The color corresponds to the phase ϕ.

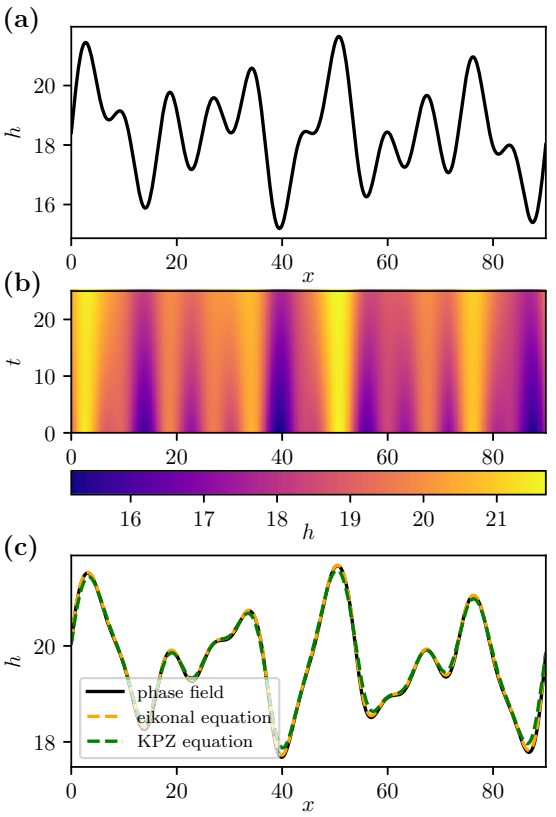

FIG. 2. (a) Initial front position obtained from the snapshot shown in Fig. 1(a) by fitting the function (2) to the data, as explained in the text. (b) Space-time dynamics of the front position, obtained from simulations of the phase field system (1). (c) Front position after integrating for $T = 25$ dimensionless time units using the phase field system and extracting the front (black solid curve), using the eikonal equation (5) (orange dashed curve) and using the KPZ equation (6) (green dashed curve).

Finally, we exploit the analytically available closed-form equations (such as the KPZ mentioned above) beyond their region of validity and showcase the data-driven learning of corrections that rectify their (now inaccurate) predictions. Making use of (and correcting) the already derived physics is important in enhancing the interpretability of the models resulting from data-driven system identification.

II. INTERFACE MODELS FOR PHASE FIELD SYSTEMS

We illustrate our approach on the phase field system described by

$$\frac{\partial \phi}{\partial t} = D\nabla^2\phi - (\phi - a)(\phi^2 - 1), \tag{1}$$

with the phase $\phi \in [-1, 1]$ and the parameters $a = -0.1$ and $D = 0.1$. This equation is integrated numerically on a two-dimensional square domain of length $L = 90$ with 400 pixels in each direction (see Sec. VI). A representative initial snapshot is shown in Fig. 1(a) where the color encodes the phase field variable ϕ. As boundary conditions, we use zero-flux boundaries $\partial\phi/\partial y = 0$ at $y = 0$ and $y = L$ and periodic boundaries $\phi(x = 0) = \phi(x = L)$ at the left and right boundaries.

The snapshot resulting from numerically integrating Eq. (1) for $T = 25$ dimensionless time units is shown in Fig. 1(b). There one can visually observe how the governing phase field system leads to a smoothening of the interface between the two phases at $\phi = 1$ and -1. This is better visualized by tracking the y position of the phase front. In order to achieve sharp interface contrast, we obtain the y positions, or height h, of the fronts from the simulation data by fitting

$$f(y) = \tanh(cy - d) \tag{2}$$

to the data at each time step and at each of the 400 discrete x positions. The front position is then defined as $h = d/c$, which is where the $f(y)$ function crosses $\phi = 0$. The height h of the front obtained this way for the initial snapshot shown in Fig. 1(a) is plotted in Fig. 2(a), whereas the front position of the snapshot at $T = 25$ shown in Fig. 1(b) is depicted in Fig. 2(c) as a black solid curve. In Fig. 2(b) the space-time profile of the front position, obtained from the phase field solution, is shown, where the color encodes the front height h.

For the phase field example considered here, one can actually analytically derive effective equations for the dynamics of the front position h [30]. First, assuming that the variations

in the direction transverse to the motion are slower than in the longitudinal direction, at the sharp interface limit [30] one can obtain the eikonal equation for the normal velocity of the interface

$$v_n = -\sqrt{2D}a + D\kappa, \tag{3}$$

with κ and v_n given by

$$\kappa = \frac{1}{\left[1+\left(\frac{\partial h}{\partial x}\right)^2\right]^{3/2}}\frac{\partial^2 h}{\partial x^2}, \quad v_n = \frac{1}{\sqrt{1+\left(\frac{\partial h}{\partial x}\right)^2}}\frac{\partial h}{\partial t} \tag{4}$$

when written in Cartesian coordinates $(x, h(x))$. This in turn means that the dynamics of the height h of the front in the phase field model (1) discussed above can be described by the equation

$$\begin{aligned}\frac{\partial h}{\partial t} &= \frac{D}{1+\left(\frac{\partial h}{\partial x}\right)^2}\frac{\partial^2 h}{\partial x^2} - \sqrt{2D}a\sqrt{1+\frac{1}{2}\left(\frac{\partial h}{\partial x}\right)^2} \\ &= f_{\text{eik}}\left(h, \frac{\partial h}{\partial x}, \frac{\partial^2 h}{\partial x^2}\right).\end{aligned} \tag{5}$$

If, in addition, $|\partial h/\partial x| \ll 1$, then the dynamics can further be approximated by Taylor expanding the right-hand side of Eq. (5), which, after disregarding cubic and higher-order terms, leads to the deterministic KPZ equation

$$\begin{aligned}\frac{\partial \tilde{h}}{\partial t} &= D\frac{\partial^2 \tilde{h}}{\partial x^2} - a\sqrt{\frac{D}{2}}\left(\frac{\partial \tilde{h}}{\partial x}\right)^2 \\ &= f_{\text{KPZ}}\left(\tilde{h}, \frac{\partial \tilde{h}}{\partial x}, \frac{\partial^2 \tilde{h}}{\partial x^2}\right),\end{aligned} \tag{6}$$

with $\tilde{h} = h + \sqrt{2D}at$.

We can now compare the dynamics of the front obtained from the phase field model (1) to the dynamics of the front predicted by the eikonal and KPZ equations. The predictions of the evolution of the initial front profile shown in Fig. 2(a) using the two interface models and periodic boundary conditions are plotted in Fig. 2(c) as an orange dashed curve and green dashed curve, respectively. In addition, the ground-truth front position obtained by integrating the two-dimensional phase field system and subsequently extracting the front position is shown as a black solid curve. Note that there is a very close correspondence between the front position at $T = 25$ obtained from the phase field model and obtained by integrating the eikonal model (5). The predictions of the KPZ deviate slightly in places where the curvature of the interface is large.

III. BLACK BOX FRONT DYNAMICS

For many applications and in broad parameter regimes, an analytic reduction to a reduced model of the front dynamics might not be possible. In such cases, we can learn the dynamics of the front in a data-driven way, for example, by learning a partial differential equation represented by a neural network [33–36]. We do this here by integrating $N_{\text{train}} = 20$ different initial conditions in the two-dimensional space domain and subsequently extracting the front position as described above. This gives 20 space-time trajectories of h on a one-dimensional spatial domain. We subsequently calculate the time derivative of the front $\partial h/\partial t$ and the space derivatives $\partial h/\partial x$ and $\partial^2 h/\partial x^2$ at each point in space and time for each trajectory using finite differences. We then use a fully connected neural network NN_Θ with weights Θ to represent the function

$$\widehat{\frac{\partial h}{\partial t}} = NN_\Theta\left(h, \frac{\partial h}{\partial x}, \frac{\partial^2 h}{\partial x^2}\right). \tag{7}$$

The weights are optimized by minimizing the mean-square error between the output of the neural network $\widehat{\partial h/\partial t}$ and the actual time derivatives of the h field (see Sec. VI for details on the training). Here we choose spatial derivatives up to second order as input to the model. How to determine the actual number of derivatives needed is beyond the scope of this work; however, approaches exist that tackle this problem, for example, by using automatic relevance determination [44]. After training this neural network, one obtains a partial differential equation, which, given proper initial conditions and boundary conditions, one can employ for simulation. The predictions of this one-dimensional front neural network model using periodic boundary conditions, as well as the predictions of the analytical PDEs, by integrating the initial front given in Fig. 2(a), are shown in Fig. 3. Note that there is good agreement between the ground-truth front dynamics (black solid curve) and the data-driven PDE (red dotted curve); however, due to the analytical approximations and the finite values of a and D, the closed-form eikonal and KPZ PDE predictions deviate slightly.

This becomes more obvious when visualizing the absolute difference between the ground-truth front position obtained from the phase field and the predictions of the surrogate models over space and time. For the eikonal equation (5), the absolute difference is shown in Fig. 4(a), with the absolute prediction error of the KPZ equation (6), presented in Fig. 4(b), where yellow indicates large deviations. In contrast,

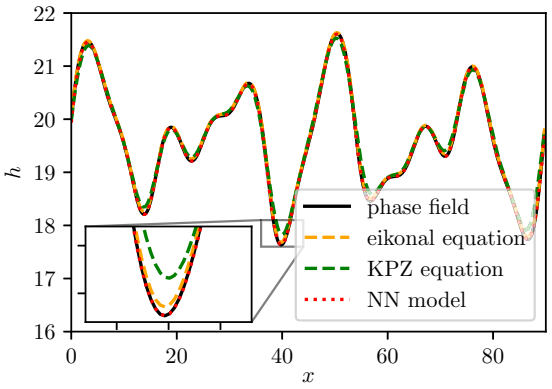

FIG. 3. Front position at $T = 25$ obtained by integrating the initial condition shown in Fig. 1(a) using the phase field system and extracting the front (black solid curve), integrating the corresponding front as shown in Fig. 2(a) using the eikonal equation (5) (orange dashed curve), using the KPZ equation (6) (green dashed curve), and using the learned PDE model (7) (red dotted curve). The inset shows an enlargement of parts of the front position, with the absolute errors shown in Fig. 4. The parameters are $a = -0.1$ and $D = 0.1$.

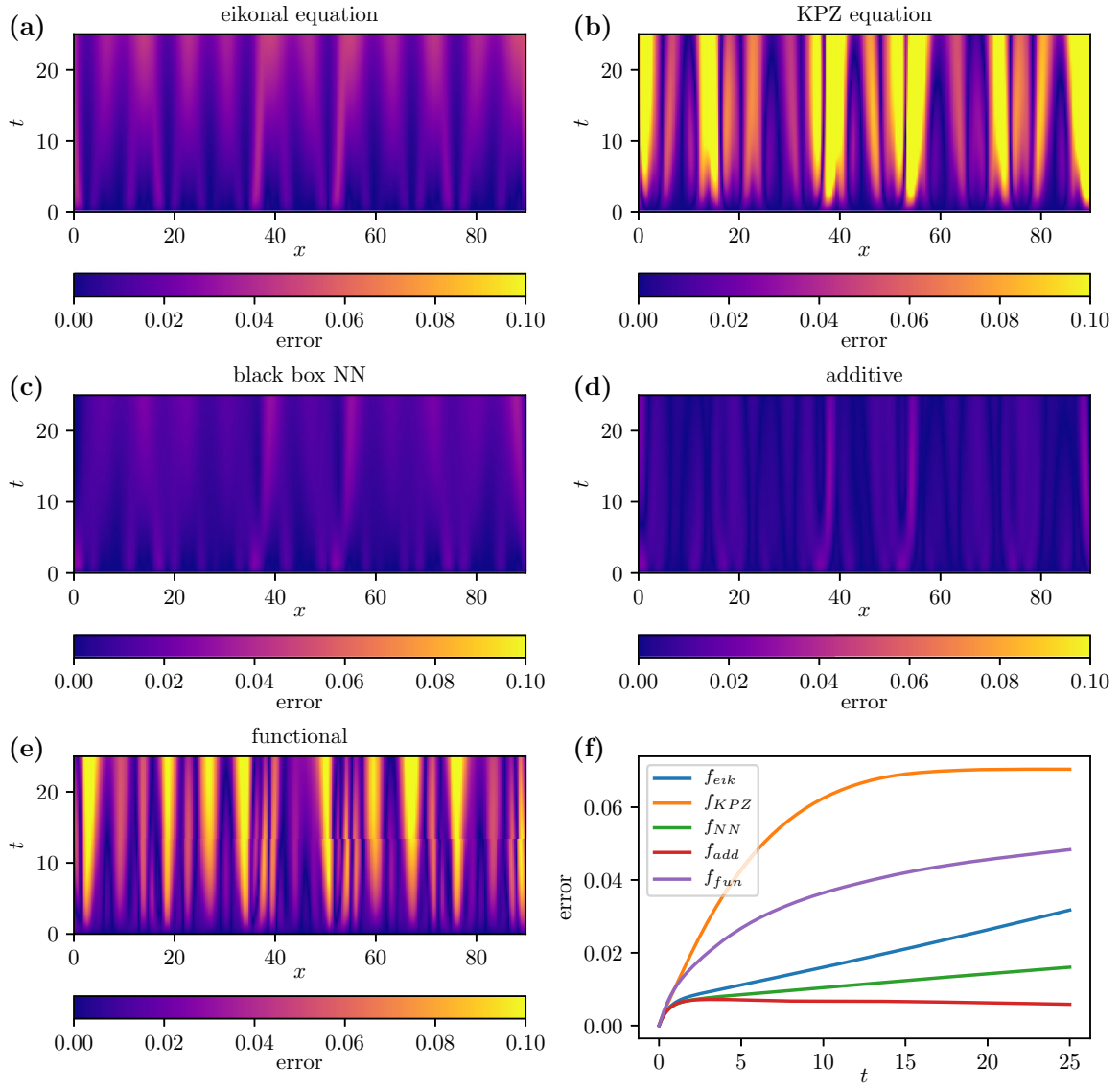

FIG. 4. Absolute error at each point in space and time between the actual front position obtained from the phase field model (the test data; see Sec. VI) and the predictions obtained by integrating the initial front snapshot from the test data using the different modeling approaches. (a) Absolute error when integrating using the analytical eikonal equation (5). (b) Absolute error when integrating using the analytical KPZ equation (6). (c) Absolute error when integrating using the data-driven black box PDE model (7). (d) Absolute error when integrating using the additive gray-box PDE model (8). (e) Absolute error when integrating using the functional gray-box PDE model (9). (f) Spatially averaged absolute error over time of the different modeling approaches. For integration, SCIPY's implementation of a Runge-Kutta 4(5) stepper [45,46] was used.

the error of the predictions obtained from the learned black box neural network PDE remains small over the time interval considered [see Fig. 4(c)].

We also investigate if the learned neural network PDE model generalizes to regions not covered by the training data. In order to do so, we predict, starting from an initial front profile at $t = 0$, for longer time intervals contained in the training data (in the training data, trajectories range until $T = 25$; see Sec. VI). In particular, we integrate the initial front profile using the original phase field system, and the learned PDE model, until $t = 50$. The resulting front profiles are depicted in Fig. 5, together with the initial front and the true front at $t = 25$ for comparison. Although the PDE model was not trained for snapshots after $t = 25$, one can observe a good correspondence between the true dynamics and the predicted dynamics of the front even for longer times.

IV. GRAY BOX FRONT DYNAMICS

In the preceding section we learned a neural network PDE description that, given the interface height h as well as the spatial partial derivatives of h, predicts the time derivative $\partial h/\partial t$ using a large set of learned weights and nonlinear activation functions. This makes such a PDE useful for prediction, yet barely accessible for interpretation, meaning that physical processes responsible for the dynamics, such as advection and diffusion, cannot be easily observed or disentangled from the resulting PDE model. In contrast, having an approximate

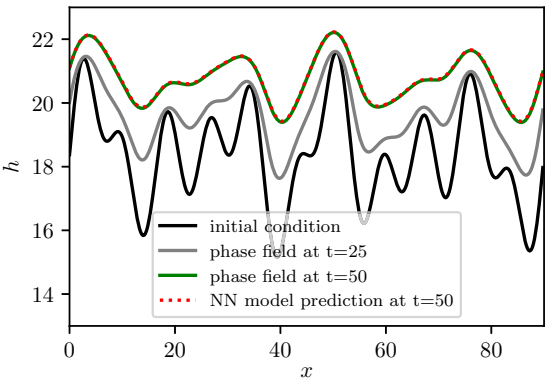

FIG. 5. Front position at $t = 0$ (black solid curve), $t = 25$ (gray solid curve), and $t = 50$ (green solid curve), the latter two obtained by integrating the initial condition at $t = 0$ [as also shown in Fig. 1(a)] using the phase field system and extracting the front. In addition, the predicted front, obtained by integrating the initial front using the learned PDE model (7), is shown (red dotted curve). The parameters are the same as in Fig. 3.

model at hand allows one to include interpretable parts in the nonlinear system identification. In particular, instead of learning a black box PDE model using a neural network as in the preceding section, one may use a neural network to learn the correction to the approximate model. We therefore phrase our first "physics-infused" modification of the system identification task as finding a neural network that, when added to the approximate white box model, gives the correct time derivative of h at each point in space and time [47]. That is, given the KPZ equation (6), we learn a neural network such that

$$\begin{aligned}\widehat{\frac{\partial h}{\partial t}} &= f_{\mathrm{KPZ}}\left(h, \frac{\partial h}{\partial x}, \frac{\partial^2 h}{\partial x^2}\right) + NN_\Theta\left(h, \frac{\partial h}{\partial x}, \frac{\partial^2 h}{\partial x^2}\right) \\ &= f_{\mathrm{add}}\left(h, \frac{\partial h}{\partial x}, \frac{\partial^2 h}{\partial x^2}\right). \end{aligned} \tag{8}$$

As in the previous examples, we use a set of 20 training trajectories and optimize the model by minimizing the mean-square error between its output and $\widehat{\frac{\partial h}{\partial t}} - f_{\mathrm{KPZ}}(h, \partial h/\partial x, \partial^2 h/\partial x^2)$. We then evaluate the efficacy of this model on the initial condition shown in Fig. 2(a). The absolute error over space and time between the integration results of the learned gray box model and the front position obtained from the original phase field system is shown in Fig. 4(d). Note that the error remains very small over the time window considered, with comparable magnitude to the black box predictions shown in Fig. 4(c).

We now propose an alternative, second physics-infused modification by exploiting our knowledge of the (now inaccurate) analytical closed-form equation models. We consider a (functional) correction of the form

$$\begin{aligned}\widehat{\frac{\partial h}{\partial t}} &= NN_\Theta\left(f_{\mathrm{KPZ}}, \frac{\partial f_{\mathrm{KPZ}}}{\partial x}, \frac{\partial^2 f_{\mathrm{KPZ}}}{\partial x^2}\right) \\ &= f_{\mathrm{fun}}\left(f_{\mathrm{KPZ}}, \frac{\partial f_{\mathrm{KPZ}}}{\partial x}, \frac{\partial^2 f_{\mathrm{KPZ}}}{\partial x^2}\right). \end{aligned} \tag{9}$$

This means that we learn the dynamics based on the local values and spatial derivatives of the KPZ closure only. Including higher derivatives of the analytical closure in the model (an alternative possibility would be to include values of the analytical closure at nearby points) can be thought of as a spatial analog of the Takens embedding for dynamical systems attractor reconstruction [48,49]. In the Takens embedding, the effect of important missing variables is modeled through inclusion of short temporal histories (time delays) of the variables we can measure. In our work here the effect of important information missed by the analytical closure is modeled through the inclusion of short spatial histories, neighboring spatial profiles, or "space delays" of the analytical closure values: We are, in a particular sense, guided by the analytical approximate closure, making the equation "higher order in space." The prediction results of this functional gray box model are shown in Fig. 4(e), which indicates a slightly inferior performance than using a simple additive correction to the KPZ.

V. DISCUSSION

We discussed and demonstrated the possibility to learn effective partial differential equations for the interface dynamics of phase field systems. This allows the reduction of the dimension of the problem at hand, facilitating tasks such as prediction or bifurcation analysis.

Our illustrative example was a model phase field system, where analytic reductions to the phase field dynamics exist. In particular, an eikonal equation (5) and an approximation thereof, the KPZ equation (6), were derived. Here we show that learning an interface PDE for the front dynamics based on simulation data and parametrized by a fully connected neural network can lead to enhanced prediction accuracy. Having access to approximate models, we furthermore highlighted the ability to find data-driven corrections to such equations, as we demonstrated for the example of the KPZ equation. We discussed two different ways to correct the operator of the approximate model: Either one can learn an additive term that rectifies the output of the KPZ equation or one can learn a correction based on the output of the KPZ equation sampled in a small spatial neighborhood of the current point of interest. Exploiting this partial physical knowledge renders the nonlinear system identification task a gray box one (to be contrasted with the physics-uninformed black box version we started with). This enhances our understanding or interpretation [50] of the predictions of the learned dynamical system: The white part of the gray box models corresponds to the KPZ equation, while only the higher-order corrections are being learned by the black box neural network, eventually making the model more easily accessible to human interpretation.

The performance of the different models discussed here was summarized in the mean prediction error over time as shown in Fig. 4(f). The KPZ shows, as expected, a performance inferior to the eikonal equation, from which it is derived via supplementary approximations. In contrast, black and gray box neural network models show superior or comparable prediction accuracy to the eikonal equation.

Different aspects, however, can influence the performance of the learned interface PDEs.

(i) The generation of the training data involves fitting the front position as well as estimating time and space derivatives (e.g., using finite differences or spectral interpolation). These steps impose numerical inaccuracies which may deteriorate the performance of the neural network PDEs.

(ii) The performance of neural networks in general depends on the choice of hyperparameters such as learning rate during training, the number of hidden layers, the optimizer, and, as in our case, the number of partial derivatives used as input to the models. A thorough investigation of these different factors might further increase the accuracy of the proposed models.

In the model phase field example considered, obtaining the phase field interface was comparatively simple (here we fitted a tanh function at each x point in space). For other examples, estimating the front position (and its movement) is a more involved, nontrivial task. Examples include the movement of cells, where the tracking of the cell boundary is a problem that attracts increasing research efforts [51]. Finding surrogate interface PDE models (and stochastic PDE models [52]) for such problems, in particular with respect to biological applications, such as cell deformation and migration processes during morphogenesis, wound healing, and cancer metastasis, is left for future work.

VI. METHODS

Neural network training

All neural networks presented in this article are composed of four layers with 96 neurons in each layer. Each hidden layer is followed by a Swish nonlinear activation function [53]. The neural networks take as input h as well as the first two spatial derivatives $\partial h/\partial x$ and $\partial^2 h/\partial x^2$. The spatial derivatives are calculated using a finite-difference stencil of length $l = 5$ and the respective finite-difference kernel for each spatial derivative of the highest accuracy order that fits into $l = 5$. As the weight initializer and the other hyperparameters we take the default settings from PYTORCH [54]. The weights are optimized using the mean-square error between the network output and the objective using the Adam optimizer [55]. As training data, we sample 20 different initial conditions, each initial condition composed of a superposition of four sinus modes with random amplitudes, taken uniformly from the interval [0, 1] and discrete frequencies, taken randomly from $2\pi/L\{0, 1, \ldots, 31, 32\}$. This enforces profiles that are periodic in x and offer a large variety. Finally, an offset drawn from the interval [10, 20] is added to the created profiles.

To generate training data, the initial profiles h created this way are mapped to two-dimensional phase field profiles ϕ. This is done by defining the phase as

$$\phi = \tanh[(h - y)/\sqrt{2D}]. \tag{10}$$

Finally, these 20 phase profiles are integrated forward using the phase field model for $T = 25$ dimensionless time units using SCIPY's implementation of a Runge-Kutta 4(5) stepper [45,46]. At 500 equidistant time steps between 0 and T, we calculate the front position as described previously in the text. Finally, we calculate the space and time derivatives at each point in space and time of these 20 trajectories, resulting in the training data for the PDE models. In addition, we test our models on the additional trajectory shown in Fig. 1 which is not contained in the training data, leading to a total of 20 training trajectories and one test trajectory.

The code to reproduce all results is available at [56].

ACKNOWLEDGMENTS

The work of F.P.K. and I.G.K. was partially supported by the U.S. Department of Energy, Grant No. SA22-0052-S001 and the U.S. Air Force Office of Scientific Research, Grant No. A9550-21-1-0317. T.M. and C.B. acknowledge financial support from the Deutsche Forschungsgemeinschaft Project No. 318763901–SFB1294. B.E. acknowledges financial support from MICINN/AEI through research Grant No. PID2020-116927RB-C22.

[1] G. J. Fix, *Free Boundary Problems* (Birkhäuser, Basel, 2004).
[2] J. W. Cahn and J. E. Hilliard, Free energy of a nonuniform system. I. Interfacial free energy, J. Chem. Phys. **28**, 258 (1958).
[3] S. M. Allen and J. W. Cahn, Ground state structures in ordered binary alloys with second neighbor interactions, Acta Metall. **20**, 423 (1972).
[4] S. M. Allen and J. W. Cahn, Mechanisms of phase transformations within the miscibility gap of Fe-rich Fe-Al alloys, Acta Metall. **24**, 425 (1976).
[5] S. M. Allen and J. W. Cahn, A microscopic theory for antiphase boundary motion and its application to antiphase domain coarsening, Acta Metall. **27**, 1085 (1979).
[6] Y. Li, S. Hu, X. Sun, and M. Stan, A review: Applications of the phase field method in predicting microstructure and property evolution of irradiated nuclear materials, npj Comput. Mater. **3**, 16 (2017).
[7] W. J. Boettinger, J. A. Warren, C. Beckermann, and A. Karma, Phase-field simulation of solidification, Annu. Rev. Mater. Res. **32**, 163 (2002).
[8] Z. Xu and P. Meakin, Phase-field modeling of solute precipitation and dissolution, J. Chem. Phys. **129**, 014705 (2008).
[9] C. Miehe, M. Hofacker, and F. Welschinger, A phase field model for rate-independent crack propagation: Robust algorithmic implementation based on operator splits, Comput. Methods Appl. Mech. Eng. **199**, 2765 (2010).
[10] Y. Suwa and Y. Saito, Computer simulation of grain growth in three dimensions by the phase field model and the Monte Carlo method, Mater. Trans. **46**, 1214 (2005).
[11] Y. Suwa and Y. Saito, Phase field simulation of the effect of anisotropy in grain boundary energy on growth kinetics and morphology of grain structure, Mater. Trans. **46**, 1208 (2005).
[12] A. Raina and C. Miehe, A phase-field model for fracture in biological tissues, Biomech. Model. Mechanobiol. **15**, 479 (2016).
[13] D. Shao, W.-J. Rappel, and H. Levine, Computational Model for Cell Morphodynamics, Phys. Rev. Lett. **105**, 108104 (2010).
[14] *Physical Models of Cell Motility*, edited by I. S. Aranson, (Springer, Cham, 2016).

[15] S. Alonso, M. Stange, and C. Beta, Modeling random crawling, membrane deformation and intracellular polarity of motile amoeboid cells, PLoS One **13**, e0201977 (2018).

[16] E. Moreno, S. Flemming, F. Font, M. Holschneider, C. Beta, and S. Alonso, Modeling cell crawling strategies with a bistable model: From amoeboid to fan-shaped cell motion, Physica D **412**, 132591 (2020).

[17] A. M. Rosende, Phase-field modeling and isogeometric analysis of cell crawling, Ph.D. thesis, Universidade da Coruña, 2017.

[18] S. Najem and M. Grant, Phase-field model for collective cell migration, Phys. Rev. E **93**, 052405 (2016).

[19] E. A. B. F. Lima, R. C. Almeida, and J. T. Oden, Analysis and numerical solution of stochastic phase-field models of tumor growth, Numer. Meth. Part. D. E. **31**, 552 (2014).

[20] T. Takaki, Phase-field modeling and simulations of dendrite growth, ISIJ Int. **54**, 437 (2014).

[21] T. Iskratsch, H. Wolfenson, and M. P. Sheetz, Appreciating force and shape—The rise of mechanotransduction in cell biology, Nat. Rev. Mol. Cell Biol. **15**, 825 (2014).

[22] H. Gomez, M. Bures, and A. Moure, A review on computational modelling of phase-transition problems, Philos. Trans. R. Soc. A **377**, 20180203 (2019).

[23] G. Caginalp and W. Xie, An analysis of phase-field alloys and transition layers, Arch. Ration. Mech. Anal. **142**, 293 (1998).

[24] Y. Kuramoto, Diffusion-induced chaos in reaction systems, Prog. Theor. Phys. Suppl. **64**, 346 (1978).

[25] G. I. Sivashinsky, Nonlinear analysis of hydrodynamic instability in laminar flames—I. Derivation of basic equations, Acta Astronaut. **4**, 1177 (1977).

[26] G. Schneider and C. E. Wayne, *Counter-Propagating Waves on Fluid Surfaces and the Continuum Limit of the Fermi-Pasta-Ulam Model* (World Scientific, Singapore, 2000), pp. 390–404.

[27] G. Schneider, Bounds for the nonlinear Schrödinger approximation of the Fermi-Pasta-Ulam system, Appl. Anal. **89**, 1523 (2010).

[28] A. Karma and W.-J. Rappel, Quantitative phase-field modeling of dendritic growth in two and three dimensions, Phys. Rev. E **57**, 4323 (1998).

[29] R. F. Almgren, Second-order phase field asymptotics for unequal conductivities, SIAM J. Appl. Math. **59**, 2086 (1999).

[30] K. R. Elder, M. Grant, N. Provatas, and J. M. Kosterlitz, Sharp interface limits of phase-field models, Phys. Rev. E **64**, 021604 (2001).

[31] G. Boussinot and E. A. Brener, Interface kinetics in phase-field models: Isothermal transformations in binary alloys and step dynamics in molecular-beam epitaxy, Phys. Rev. E **88**, 022406 (2013).

[32] M. Kardar, G. Parisi, and Yi-C. Zhang, Dynamic Scaling of Growing Interfaces, Phys. Rev. Lett. **56**, 889 (1986).

[33] R. González-García, R. Rico-Martínez, and I. G. Kevrekidis, Identification of distributed parameter systems: A neural net based approach, Comput. Chem. Eng. **22**, S965 (1998).

[34] K. Krischer, R. Rico-Martínez, I. G. Kevrekidis, H. H. Rotermund, G. Ertl, and J. L. Hudson, Model identification of a spatiotemporally varying catalytic reaction, AIChE J. **39**, 89 (1993).

[35] R. Rico-Martínez, K. Krischer, I. G. Kevrekidis, M. C. Kube, and J. L. Hudson, Discrete- vs. continuous-time nonlinear signal processing of Cu electrodissolution data, Chem. Eng. Commun. **118**, 25 (1992).

[36] F. P. Kemeth, T. Bertalan, T. Thiem, F. Dietrich, S. J. Moon, C. R. Laing, and I. G. Kevrekidis, Learning emergent partial differential equations in a learned emergent space, Nat. Commun. **13**, 3318 (2022).

[37] 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. USA **113**, 3932 (2016).

[38] 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).

[39] Z. Long, Y. Lu, X. Ma, and B. Dong, in *Proceedings of the 35th International Conference on Machine Learning, Stockholm, 2018*, edited by J. Dy and A. Krause (PMLR, La Jolla, 2018), Vol. 80, pp. 3208–3216.

[40] L. Lu, P. Jin, G. Pang, Z. Zhang, and G. E. Karniadakis, Learning nonlinear operators via deeponet based on the universal approximation theorem of operators, Nat. Mach. Intell. **3**, 218 (2021).

[41] Z. Li, N. B. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar, Fourier Neural Operator for Parametric Partial Differential Equations, in *International Conference on Learning Representations* (ICLR, La Jolla, 2021).

[42] M. Raissi, P. Perdikaris, and G. E. 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 (2019).

[43] E. Kiyani, S. Silber, M. Kooshkbaghi, and M. Karttunen, Machine learning based data-driven discovery of nonlinear phase-field dynamics, Phys. Rev. E **106**, 065303 (2022).

[44] S. Lee, M. Kooshkbaghi, K. Spiliotis, C. I. Siettos, and I. G. Kevrekidis, Coarse-scale PDEs from fine-scale observations via machine learning, Chaos **30**, 013141 (2020).

[45] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey *et al.* (SciPy 1.0 Contributors), SciPy 1.0: Fundamental algorithms for scientific computing in Python, Nat. Methods **17**, 261 (2020).

[46] J. R. Dormand and P. J. Prince, A family of embedded Runge-Kutta formulae, J. Comput. Appl. Math. **6**, 19 (1980).

[47] S. Pan and K. Duraisamy, Data-driven discovery of closure models, SIAM J. Appl. Dyn. Syst. **17**, 2381 (2018).

[48] H. Whitney, Differentiable manifolds, Ann. Math. **37**, 645 (1936).

[49] F. Takens, *Detecting Strange Attractors in Turbulence* (Springer, Berlin, 1981), pp. 355–381.

[50] C. Molnar, *Interpretable Machine Learning* (Molnar, Munich, 2022).

[51] D. Schindler, T. Moldenhawer, M. Stange, V. Lepro, C. Beta, M. Holschneider, and W. Huisinga, Analysis of protrusion dynamics in amoeboid cell motility by means of regularized contour flows, PLoS Comput. Biol. **17**, e1009268 (2021).

[52] F. Dietrich, A. Makeev, G. Kevrekidis, N. Evangelou, T. Bertalan, S. Reich, and I. G. Kevrekidis, Learning effective stochastic differential equations from microscopic simulations: Combining stochastic numerics and deep learning, arXiv:2106.09004.

[53] P. Ramachandran, B. Zoph, and Q. V. Le, Swish: A self-gated activation function, arXiv:1710.05941.

[54] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai *et al.*, in *Advances in Neural Information Processing Systems 32*, edited by F. d'Alché Buc, E. Fox, and R. Garnett (Curran, Red Hook, 2019), pp. 8024–8035.

[55] D. P. Kingma and J. Ba, in *Proceedings of the 3rd International Conference on Learning Representations, San Diego, 2015*, edited by Y. Bengio and Y. LeCun (ICLR, La Jolla, 2017).

[56] https://github.com/fkemeth/front_pde.