Machine-Learning Non-Conservative Dynamics for New-Physics Detection

Ziming Liu,[1,2,3,*] Bohan Wang,[1] Qi Meng,[1] Wei Chen,[1,†] Max Tegmark,[2,3,‡] and Tie-Yan Liu[1,§]

[1] *Microsoft Research Asia, Beijing, China*
[2] *Department of Physics, Massachusetts Institute of Technology, Cambridge, USA*
[3] *AI Institute for Artificial Intelligence and Fundamental Interactions (IAIFI)*

(Dated: June 3, 2021)

Energy conservation is a basic physics principle, the breakdown of which often implies *new physics*. This paper presents a method for data-driven "new physics" discovery. Specifically, given a trajectory governed by unknown forces, our *Neural New-Physics Detector (NNPhD)* aims to detect new physics by decomposing the force field into conservative and non-conservative components, which are represented by a Lagrangian Neural Network (LNN) and a universal approximator network (UAN), respectively, trained to minimize the force recovery error plus a constant λ times the magnitude of the predicted non-conservative force. We show that a phase transition occurs at $\lambda = 1$, universally for arbitrary forces. We demonstrate that NNPhD successfully discovers new physics in toy numerical experiments, rediscovering friction (1493) from damped double pendulum, Neptune from Uranus' orbit (1846) and gravitational waves (2017) from an inspiraling orbit. We also show how NNPhD coupled with an integrator outperforms previous methods for predicting the future of a damped double pendulum.

I. INTRODUCTION

Energy conservation is a fundamental physical law, so when non-conservation is observed, physicists often consider it evidence of an unseen body or novel external forces rather than questioning the conservation law itself. In this paper, we will therefore refer to energy non-conservation as simply *new physics* and strive to auto-detect it[1]. Many experimental new physics discoveries have manifested as apparent violation of energy conservation, for example friction [2], Neptune [3], neutrinos [4], dark matter [5, 6], extra-solar planets [7] and gravitational waves [8]. We focus on classical mechanics in this paper, but the idea extends to all fields of physics including quantum mechanics. We illustrate several classic examples in FIG. 1. In these cases, the new physics was historically identified from the residual force after fitting data to a conservative force of a *known functional form*. The key novel contribution in this paper is that our proposed model, dubbed the Neural New Physics Detector (NNPhD), can discover the new physics even when the form of the conservative "old physics" is *not known*.

Data-driven discovery has proven extremely useful in physics, yet also non-trivial. For example, Kepler spent 25 years analyzing astronomical data before formulating his eponymous three laws. In this paper, we aim to automate and accelerate data-driven new physics discovery using machine learning tools. More concretely, given the trajectory of one or several objects governed by some force, we aim to decompose the force into conservative and non-conservative parts, followed by a symbolic regression module for explanation. As a trivial example, we aim to decompose the force $f = -kq - \gamma\dot{q}$ of a damped harmonic oscillator into conservative part $f_{\rm c} = -kq$ and a non-conservative part $f_{\rm n} = -\gamma\dot{q}$.

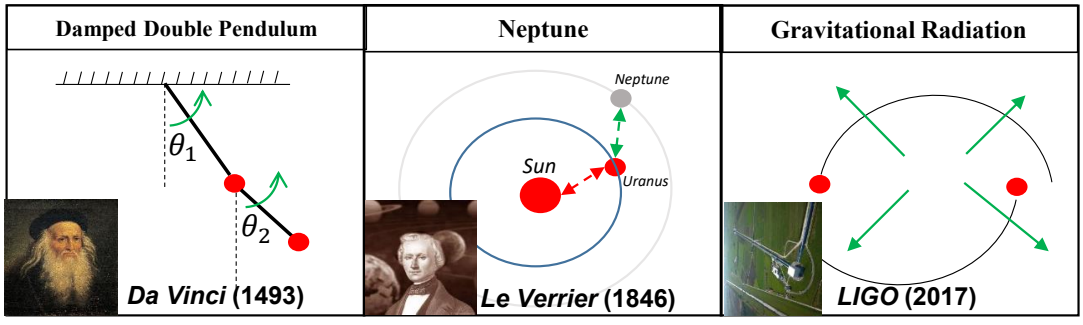


FIG. 1: NNPhD can auto-rediscover several classic examples.

* zmliu@mit.edu
† wche@microsoft.com
‡ tegmark@mit.edu
§ tyliu@microsoft.com

[1] In contrast, "new physics" in high energy physics specifically refers to "new fundamental particles" or "new fundamental interactions" beyond the Standard Model [1].

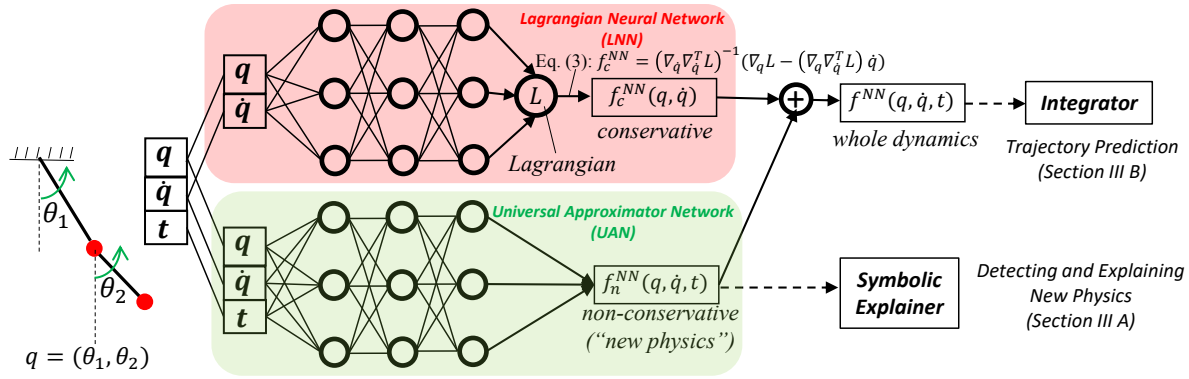


FIG. 2: NNPhD predicts dynamics by decomposing the force into conservative and non-conservative components, which can reveal new physics and improve trajectory extrapolation.

Conservation laws have been introduced into neural networks as strong inductive biases, such as in the Lagrangian Neural Network (LNN) [9], the Hamiltonian Neural Network (HNN) [10] and variants [11–14]. The limitation of these models lies in their inability to model non-conservative dynamics, where the non-conservation can be caused by dissipation, external driving forces, etc. Our proposed NNPhD can resolve this limitation by augmenting LNN with a universal approximation network (UAN), illustrated in FIG. 2. Although prior works [9–35] attempt to learn the general or conservative force from data, most of these methods are unable to perform force decomposition, except for [32, 35] which assume (partial) knowledge of physics thus lose generality. Moreover, while the current literature mostly focuses on model *predictability*, we pay extra attention to *explainability* made possible by symbolic regression.

The rest of this paper is organized as follows: In Section II, we review the problem framing and useful results, define *force decomposition with minimal non-conservation* and propose NNPhD to learn this force decomposition. In Section III, we carry out numerical experiments to verify our theoretical analysis of the presented algorithm, as well as to demonstrate the potential of NNPhD for new-physics discovery.

II. METHOD

A. Notation

We consider the general classical physical system described by an n-dimensional vector $\mathbf{q}$ of generalized coordinates whose time-evolution $\mathbf{q}(t)$ is governed by a second-order ordinary differential equation

$$\ddot{\mathbf{q}} = f(\mathbf{q}, \dot{\mathbf{q}}, t), \tag{1}$$

where $f : \mathbb{R}^{2n+1} \to \mathbb{R}^n$. The *acceleration* $\ddot{\mathbf{q}}$ is intimately related to *force* according to Newton's second law. In the following, we for simplicity refer to $f(\mathbf{q}, \dot{\mathbf{q}}, t)$ as a *force field* (dynamics perspective) or *acceleration field* (kinematics perspective) interchangeably. The dynamical systems in our numerical examples consist of k particles in d dimensions, so $n = kd$ and $\mathbf{q} \equiv [\mathbf{q}_1, \cdots, \mathbf{q}_k] \in \mathbb{R}^n$, but our NNPhD method is fully general and makes no such assumptions.

An important subset of dynamical systems are known as *conservative* because they conserve energy, which can be described by Euler-Lagrange Equation:

$$\frac{d}{dt}\nabla_{\dot{\mathbf{q}}}\mathcal{L} = \nabla_{\mathbf{q}}\mathcal{L}, \tag{2}$$

where $\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})$ is the Lagrangian and ∇ is the gradient operator. As reviewed in Appendix A and [9], the Lagrangian mechanics formalism implies that such systems allows equation (1) to be re-expressed as

$$\ddot{\mathbf{q}} = (\nabla_{\dot{\mathbf{q}}}\nabla_{\dot{\mathbf{q}}}^T\mathcal{L})^{-1}\left[\nabla_{\mathbf{q}}\mathcal{L} - (\nabla_{\mathbf{q}}\nabla_{\dot{\mathbf{q}}}^T\mathcal{L})\dot{\mathbf{q}}\right] \tag{3}$$

For readers whose background is primarily in machine learning rather than physics, Appendix D provides a brief review of the Lagrangian mechanics formalism that we use in this paper.

B. Lagrangian neural networks

To guarantee energy conservation, inductive biases have recently been embedded into neural networks, including Lagrangian Neural Network [9], Hamiltonian Neural Network [10] and variants [11–14]. As shown in FIG. 2, a LNN uses a neural network to parametrize the Lagrangian $\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})$ and output $f_c^{NN}(\mathbf{q}, \dot{\mathbf{q}})$ through evaluating Eq. (3). For a given loss function defined between model output $f_c^{NN}(\mathbf{q}, \dot{\mathbf{q}})$ and ground truth $\ddot{\mathbf{q}}$, the LNN parameters can be learned using standard optimization algorithms. A trained LNN therefore contains a Lagrangian that determines conservative dynamics. Since not all physical systems conserve energy, the Lagrangian mechanics is insufficient for describing non-conservative dynamics, motivating the NNPhD framework.

C. The force decomposition minimizing non-conservation

Following the problem setting of LNN, we focus on the simple setting where the acceleration $\ddot{\mathbf{q}}$ is a known function i.e., $\ddot{\mathbf{q}} \equiv f(\mathbf{q}, \dot{\mathbf{q}}, t)$. Our goal is therefore not to *learn* the force field, but to *decompose* the force field. In practice, where only discrete points on trajectory $\{(\mathbf{q}^{(i)}, t^{(i)})\}$ are known, $\dot{\mathbf{q}}^{(i)}$ and $\ddot{\mathbf{q}}^{(i)}$ can be extracted using a Neural ODE module [16].

The main goal is to decompose the force field $f(\mathbf{q}, \dot{\mathbf{q}}, t) : \mathbb{R}^{2n+1} \to \mathbb{R}^n$ into a (time-independent) conservative component $f_{\rm c}(\mathbf{q}, \dot{\mathbf{q}}) : \mathbb{R}^{2n} \to \mathbb{R}^n$ and a non-conservative component $f_{\rm n}(\mathbf{q}, \dot{\mathbf{q}}, t) : \mathbb{R}^{2n+1} \to \mathbb{R}^n$ such that

$$f(\mathbf{q}, \dot{\mathbf{q}}, t) = f_{\rm c}(\mathbf{q}, \dot{\mathbf{q}}) + f_{\rm n}(\mathbf{q}, \dot{\mathbf{q}}, t). \tag{4}$$

In general, the decomposition is not unique. We desire the decomposition that minimizes the non-conservative component $f_{\rm n}(\mathbf{q}, \dot{\mathbf{q}}, t)$. To define the distance between two functions, we embed all functions $f(\mathbf{q}, \dot{\mathbf{q}}, t)$ in a normed vector space $(\mathcal{F}, \|\cdot\|)$ and define its conservative subspace $\mathcal{F}_{\rm c} \subset \mathcal{F}$ as

$$\mathcal{F}_{\rm c} = \left\{ f \in \mathcal{F} \middle| \begin{array}{l} \exists \mathcal{L}(\mathbf{q},\dot{\mathbf{q}}):\mathbb{R}^{2n}\to\mathbb{R}, \quad s.t. \\ f(\mathbf{q},\dot{\mathbf{q}})=(\nabla_{\dot{\mathbf{q}}}\nabla_{\dot{\mathbf{q}}}^T\mathcal{L})^{-1}\left(\nabla_{\mathbf{q}}\mathcal{L}-(\nabla_{\mathbf{q}}\nabla_{\dot{\mathbf{q}}}^T\mathcal{L})\dot{\mathbf{q}}\right) \end{array} \right\}.$$

We formally define the force decomposition as follows:

Definition II.1. ***(force decomposition with minimal non-conservation)*** The *conservative component* of $f(\mathbf{q}, \dot{\mathbf{q}}, t)$ is defined as

$$f_{\rm c}(\mathbf{q}, \dot{\mathbf{q}}) \equiv \arg\min_{g \in \mathcal{F}_{\rm c}} \|f(\mathbf{q}, \dot{\mathbf{q}}, t) - g(\mathbf{q}, \dot{\mathbf{q}})\|. \tag{5}$$

We denote $f_{\rm n}(\mathbf{q}, \dot{\mathbf{q}}, t) \equiv f(\mathbf{q}, \dot{\mathbf{q}}, t) - f_{\rm c}(\mathbf{q}, \dot{\mathbf{q}})$ the *non-conservative component* of f and denote the decomposition $f(\mathbf{q}, \dot{\mathbf{q}}) = f_{\rm c}(\mathbf{q}, \dot{\mathbf{q}}) + f_{\rm n}(\mathbf{q}, \dot{\mathbf{q}}, t)$ the *force decomposition minimizing non-conservation*.

D. Neural New-Physics Detector (NNPhD) framework

To learn the force decomposition minimizing non-conservation, we define a learning framework dubbed the *Neural New-Physics Detector* (NNPhD). Specifically, NNPhD learns $f_{\rm c}$ and $f_{\rm n}$ jointly. As illustrated in FIG. 2, NNPhD consists of two parallel modules, a Lagrangian Neural Network (LNN) and a Universal Approximator Network (UAN). The LNN takes in $(\mathbf{q}, \dot{\mathbf{q}})$ to predict a Lagrangian $\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}; \boldsymbol{w}_{\rm c})$ in the intermediate layer and outputs $f_{\rm c}^{NN}(\mathbf{q}, \dot{\mathbf{q}}; \boldsymbol{w}_{\rm c})$ calculated from Eq. (3), where $\boldsymbol{w}_{\rm c}$ are LNN parameters. The UAN is a pure black box (a fully connected neural network) that takes in $(\mathbf{q}, \dot{\mathbf{q}}, t)$ and outputs $f_{\rm n}^{NN}(\mathbf{q}, \dot{\mathbf{q}}, t; \boldsymbol{w}_{\rm n})$ where $\boldsymbol{w}_{\rm n}$ are parameters of the black box. The two outputs are summed to predict the full force field

$$f^{NN}(\mathbf{q}, \dot{\mathbf{q}}, t; \boldsymbol{w}_{\rm c}, \boldsymbol{w}_{\rm n}) = f_{\rm c}^{NN}(\mathbf{q}, \dot{\mathbf{q}}; \boldsymbol{w}_{\rm c}) + f_{\rm n}^{NN}(\mathbf{q}, \dot{\mathbf{q}}, t; \boldsymbol{w}_{\rm n}). \tag{6}$$

We take both *recovery error* and *minimal non-conservation* into considerations to design our loss function: (1) f^{NN} should recover ground truth f; (2) we make maximal use of $f_{\rm c}^{NN}$ and reduce $f_{\rm n}^{NN}$ as much as possible (e.g. when f is conservative, we hope that $f_{\rm n}^{NN}$ vanishes). Guided by these two principles, we define our loss function as follows (denoting the i^{th} sample $\mathbf{x}^{(i)} \equiv (\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)})$):

$$\begin{aligned} &L_{NNPhD}(\boldsymbol{w}_{\rm c}, \boldsymbol{w}_{\rm n}) = L_e(\boldsymbol{w}_{\rm c}, \boldsymbol{w}_{\rm n}) + \lambda L_b(\boldsymbol{w}_{\rm n}), \\ &L_b(\boldsymbol{w}_{\rm n}) \equiv \left(\frac{1}{Nn} \sum_{i=1}^{N} ||f_{\rm n}^{NN}(\mathbf{x}^{(i)}, t^{(i)}; \boldsymbol{w}_{\rm n})||^p \right)^{\frac{1}{p}}, \\ &L_e(\boldsymbol{w}_{\rm c}, \boldsymbol{w}_{\rm n}) \equiv \\ &\left(\frac{1}{Nn} \sum_{i=1}^{N} ||f_{\rm c}^{NN}(\mathbf{x}^{(i)}; \boldsymbol{w}_{\rm c}) + f_{\rm n}^{NN}(\mathbf{x}^{(i)}, t^{(i)}; \boldsymbol{w}_{\rm n}) - f(\mathbf{x}^{(i)}, t^{(i)})||^p \right)^{\frac{1}{p}}, \end{aligned} \tag{7}$$

where $p \geq 1$ and the regularization coefficient $\lambda > 0$. The factors $\frac{1}{N}$ and $\frac{1}{n}$ average over samples and degrees of freedom, respectively. Here we use L_p function norms, i.e. $\|f\| \equiv (\int |f|^p d\mu)^{1/p}$, where the integral is replaced by averaging over finite training samples. L_e is the recovery error and L_b penalizes the black box module to discourage it from learning conservative dynamics.

E. The regularization phase transition

Does minimizing Eq. (7) yield the force decomposition of Eq. (5)? We offer an affirmative answer to this question by presenting Theorem 1 informally here. Appendix F provides a rigorous formulation and proof of this theorem.

Theorem 1. *(Informal) Suppose $f_{\rm c}^{NN}$ and $f_{\rm n}^{NN}$ can represent any conservative force field and any (continuous) force field, and $(f_{\rm c}^*, f_{\rm n}^*)$ denotes the pair that minimizes NNPhD loss from Eq. (7). Then we have a phase transition at $\lambda = 1$ such that $(f_{\rm c}^*, f_{\rm n}^*) = (f_{\rm c}, f_{\rm n})$ when $0 < \lambda < 1$, and $(f_{\rm c}^*, f_{\rm n}^*) = (f_{\rm c}, 0)$ when $\lambda > 1$.*

Theorem 1 has two interesting and useful implications: (1) *sharp phase transition*: The recovery error $L_e = 0$ when $\lambda < 1$ and $L_e = \|f_{\rm n}\| > 0$ when $\lambda > 1$. As a result, non-conservative dynamics predicts an error jump of L_e at $\lambda = 1$, while conservative dynamics does not. This phenomenon justifies the term "detector" in our model name, in the sense that non-conservative dynamics is *detected* by the sharp phase transition at $\lambda = 1$. (2) *effortless λ tuning*: Any $\lambda \in (0, 1)$ would achieve the force decomposition. Below we report numerical experiments showing that in practice, too small λ do not regularize UAN effectively, and force decomposition results are robust for $0.05 \lesssim \lambda < 1$ independent of dynamical systems at study.

As we will see in Appendix F, the proof is more complicated than one might naively expect. If conservative

force fields formed a linear subspace, then the conservative component f_c from equation (5) would simply be the orthogonal projection onto that space, and the non-conservative residual f_n would be orthogonal to that subspace. But conservative force fields as we have defined them generally do *not* form a linear subspace, *i.e.*, the sum of two energy-conserving force fields may not conserve energy, which is related to the nonlinear nature of equation (3).

III. RESULTS FROM NUMERICAL EXPERIMENTS

In this section, we test our NNPhD algorithm with a series of numerical examples defined in Table I. In Section III A, we quantify its ability to rediscover symbolic expressions for "new physics" such as friction, Neptune and gravitational waves. In Section III B, we show that, although NNPhD is designed for new physics detection, it can also outperform baseline trajectory prediction for the damped double pendulum example. In Section III C, we use toy examples to verify and quantify the aforementioned λ-dependent phase transition, and explore how the choices of p and λ in Eq. (7) influence algorithm behavior. Finally we discuss how data quality affects identifiability of new physics in Section III D. Further technical details on model parameters, simulations and neural network architecture are provided in Appendix A.

A. Discovery of New Physics

We now test NNPhD on three numerical examples defined in Table I, to see if it can rediscover friction (1493), Neptune (1846) and gravitational wave emission (2017). In all three cases, the force fields defined by the right hand side are the sum of a conservative part (the first term) and a non-conservative "new physics" part (the second term) that we hope to discover. Before delving into our numerical experiments, let us briefly comment on how we model these three dynamical systems.

1. Physical systems tested

Friction Italian polymath Leonardo da Vinci first recorded the basic laws of friction in 1493. We add friction to the double pendulum system and to test if NNPhD can automatically discover the friction force solely from data. The damped double pendulum example can be described by two angles and their derivatives *i.e.*, $\mathbf{q} = (\theta_1, \theta_2)$ and $\dot{\mathbf{q}} = (\dot{\theta}_1, \dot{\theta}_2)$. In our numerical experiment, we choose the physical parameters $m_1 = m_2 = g = l_1 = l_2 = 1$, $\gamma = 0.02$.

Neptune *Le Verrier* postulated the existence of Neptune in 1846: astronomers had found that Uranus' orbit around the Sun precessed in a way suggesting the presence of a force of unknown cause, later identified as Neptune. Neptune was invisible at the time in the sense that contemporary astronomers could not observe its position or velocity, but Le Verrier (and NNPhD) were able to identify the existence of a third body by identifying a non-conservative contribution to the force field of the two-body system. For our numerical experiments, we make the simplifying assumptions that (1) the Sun remains fixed at the origin, (2) the elliptical orbits of Uranus and Neptune are circular (have eccentricity $e = 0$) and lie in the same plane, (3) Neptune's orbit is unaffected by Uranus, and (4) the effects of other planets are negligible. Here x and y denote the coordinates of Uranus, and time t is measured in units such that Uranus' orbital period is $2\pi\sqrt{2^3}$. We choose $G = 1$, mass of Sun $M_\odot = 1$. Neptune's mass, orbital radius and angular velocity are $M_n = 0.005$, $r_n = 3$ and $\omega_n = 3^{-\frac{3}{2}} \approx 0.192$.

Gravitational Radiation As predicted by Einstein, the gravitational two-body problem is non-conservative, since the system radiates gravitational radiation that carries away energy and causes orbital decay. Experimental confirmation of this garnered Nobel Prizes both in 1993 (for the Hulse-Taylor pulsar) and in 2017 (for the LIGO discovery of gravitational waveforms from black hole mergers), and there is great current interest in exploiting such signals both for gravitational wave astronomy and for precision tests of general relativity. To test whether NNPhD can auto-discover the non-conservative force caused by gravitational wave back-reaction solely from black hole trajectories, we simulate a binary black hole inspiral using the approximation from [36] that the radiated gravitational wave power $P = \frac{32}{5}\frac{G}{c^5}\mu^2 r^4 \Omega^6 = \frac{32}{5}\frac{G}{c^5}\mu^2 \frac{v^6}{r^2}$ in a slowly decaying circular orbit (of radius r, angular frequency Ω and reduced mass $\mu \equiv (M_1^{-1} + M_2^{-1})^{-1}$) equals the energy loss rate $-dE/dt = vf$ from a dissipative back-reaction force f. Using $\Omega \propto r^{-3/2}$ and $v \propto r^{-1/2}$ from Kepler's 3rd law gives $P \propto v^{10}$, with a total force

$$\boldsymbol{f} = \mu\ddot{\boldsymbol{r}} = -\frac{GM_1M_2}{r^3}\boldsymbol{r} - \frac{32M_1^2M_2^2(M_1^2+M_2^2)}{5Gc^5(M_1+M_2)^6}v^8\boldsymbol{v}, \quad (8)$$

corresponding to an acceleration

$$\ddot{\boldsymbol{r}} = -\frac{G(M_1+M_2)}{r^3}\boldsymbol{r} - \frac{32M_1M_2(M_1^2+M_2^2)}{5Gc^5(M_1+M_2)^5}v^8\boldsymbol{v}, \quad (9)$$

where $\boldsymbol{r} = \boldsymbol{r}_2 - \boldsymbol{r}_1$ and $\boldsymbol{v} = \boldsymbol{v}_2 - \boldsymbol{v}_1$. We choose these physical parameters to be $G = M_1 = M_2 = 1, c = 3$.

2. Detection of new physics

These three physical systems have $d = 2$ degrees of freedom, obeying the second-order coupled differential equations in Table I. Including the corresponding conjugate momenta, a system's state is thus a point moving along some trajectory in a $2d$-dimensional phase space,

Model	Equation
Damped Double Pendulum	$\begin{pmatrix}\ddot{\theta}_1\\ \ddot{\theta}_2\end{pmatrix} = \begin{pmatrix}\frac{m_2 l_1 \dot{\theta}_1^2 \sin(\theta_2-\theta_1)\cos(\theta_2-\theta_1)+m_2 g \sin\theta_2 + m_2 l_2 \dot{\theta}_2^2 \sin(\theta_2-\theta_1) - (m_1+m_2) g \sin\theta_1}{(m_1+m_2) l_1 - m_2 l_1 \cos^2(\theta_2-\theta_1)}\\ \frac{-m_2 l_2 \dot{\theta}_2^2 \sin(\theta_2-\theta_1) + (m_1+m_2)(g\sin\theta_1 \cos(\theta_2-\theta_1) - l_1 \dot{\theta}_1^2 \sin(\theta_2-\theta_1) - g\sin\theta_2)}{(m_1+m_2) l_1 - m_2 l_1 \cos^2(\theta_2-\theta_1)}\end{pmatrix} - \gamma \begin{pmatrix}\dot{\theta}_1\\ \dot{\theta}_2\end{pmatrix}$
Neptune	$\begin{pmatrix}\ddot{x}\\ \ddot{y}\end{pmatrix} = \begin{pmatrix}-\frac{GM_\odot x}{(x^2+y^2)^{\frac{3}{2}}}\\ -\frac{GM_\odot y}{(x^2+y^2)^{\frac{3}{2}}}\end{pmatrix} + \begin{pmatrix}\frac{GM_n(-x+r_n\cos(\omega_n t))}{[(x-r_n\cos(\omega_n t))^2+(y-r_n\sin(\omega_n t))^2]^{\frac{3}{2}}}\\ \frac{GM_n(-y+r_n\sin(\omega_n t))}{[(x-r_n\cos(\omega_n t))^2+(y-r_n\sin(\omega_n t))^2]^{\frac{3}{2}}}\end{pmatrix}$
Gravitational Radiation	$\begin{pmatrix}\ddot{x}\\ \ddot{y}\end{pmatrix} = \begin{pmatrix}-\frac{G(M_1+M_2)x}{(x^2+y^2)^{\frac{3}{2}}}\\ -\frac{G(M_1+M_2)y}{(x^2+y^2)^{\frac{3}{2}}}\end{pmatrix} + \frac{32 M_1 M_2 (M_1^2+M_2^2)}{5Gc^5(M_1+M_2)^5}\begin{pmatrix}-(\dot{x}_i^2+\dot{y}_i^2)^4 \dot{x}_i\\ -(\dot{x}_i^2+\dot{y}_i^2)^4 \dot{y}_i\end{pmatrix}$

TABLE I: We test if NNPhD can automatically decompose these three force fields into a conservative part (first term) and a non-conservative part (second term) corresponding to the "new physics".

satisfying a $2d$ coupled first-order coupled differential equations. We solve these equations and compute the trajectories numerically using a 4th-order Runge-Kutta integrator at $N_{step} = \{300, 1000, 300\}$ timesteps of size $\varepsilon = \{0.1, 0.1, 0.05\}$ for the three physical systems, using the following initial conditions:

$$\begin{aligned}(\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2) &= (1, 0, 0, 0)\\ (x, y, \dot{x}, \dot{y}) &= (3, 0, 0, \frac{1}{\sqrt{3}})\\ (x, y, \dot{x}, \dot{y}) &= (0, 2, -1, 0)\end{aligned} \tag{10}$$

Once trajectory points are calculated, the ground truth forces f at those points are evaluated using the formula in TABLE I [2]. We do not hold back any testing data in this section, since many insights can be gained solely from training data. We will hold back testing data and verify NNPhD's generalization ability in Section III B.

We then train NNPhD on the aforementioned trajectory data as detailed in Appendix B. FIG. 3 shows the resulting NNPhD prediction loss L_e as a function of λ, revealing a striking phase transition at $\lambda = 1$: for $\lambda < 1$, L_e is almost zero, while for $\lambda > 1$, L_e is an approximately constant positive number, indicating the magnitude of non-conservative components.

As we showed above, such a phase transition is a smoking-gun signature of new physics manifesting as non-conservative dynamics. The observed phase transitions thus justify the NNPhD name.

3. *Modeling of New Physics with Symbolic Expressions*

After detecting the existence of new physics, physicists are interested in understanding and explaining this new

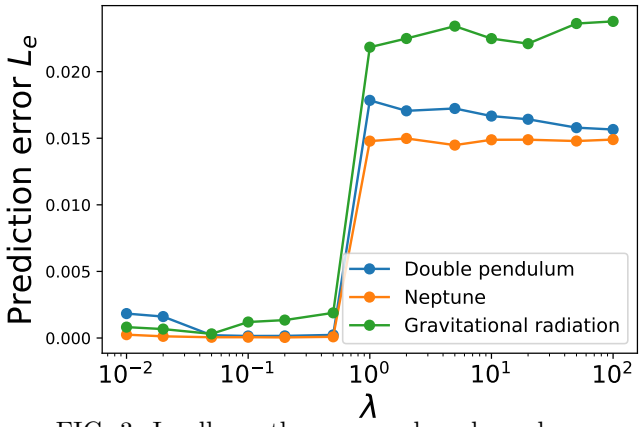


FIG. 3: In all our three examples, clear phase transitions at $\lambda = 1$ indicate the existence of new physics.

physics by describing it with via symbolic expressions. We found that if we did not impose any inductive biases on the LNN, we unfortunately did not auto-discover ant meaningful symbolic expressions. We therefore drew inspiration from the history of physics, where inductive biases have routinely been used. For example, physicists often knew and used analytic formulas for the old physics when quantifying new physics. In this spirit, we constrain the form of LNN Lagrangian so that only a set of coefficients are learnable, while the UAN remains to a fully general feedforward neural network with two hidden layers containing 200 neurons each. Specifically, we parametrize the Lagrangians for our three examples as follows:

$$\begin{aligned}\mathcal{L}_{\text{fric}} &= c_1\cos\theta_1 + c_2\cos\theta_2 + c_3\dot{\theta}_1^2 + c_4\dot{\theta}_2^2 + c_5\dot{\theta}_1\dot{\theta}_2\cos(\theta_1-\theta_2)\\ \mathcal{L}_{\text{neptune}} &= c_1\dot{x}^2 + c_2\dot{y}^2 + \frac{c_3}{\sqrt{x^2+y^2}}\\ \mathcal{L}_{\text{grav}} &= c_1\dot{x}^2 + c_2\dot{y}^2 + \frac{c_3}{\sqrt{x^2+y^2}}\end{aligned} \tag{11}$$

This is implemented by inputting hand-crafted features ($\cos\theta, x^2$, *etc.*) into a learnable linear layer which outputs the predicted Lagrangian. We adopt a train-and-explain strategy:

1. **Training:** Like before, we train the whole NNPhD (LNN and UAN are updated simultaneously) with $\lambda = 0.2$ using the ADAM optimizer with annealing learning rate $\{10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}\}$ for 2000

[2] In more realistic settings, one would first extract $\ddot{\mathbf{q}}$ from trajectory data, e.g. with Neural ODE [16] or AI Physicist [27], and then use $\ddot{\mathbf{q}}$ as labels to train NNPhD. We treat $\ddot{\mathbf{q}} = f(\mathbf{q}, \dot{\mathbf{q}}, t)$ as an oracle in this paper since we focus on the force field decomposition aspect.

TABLE II: Symbolic Formulas Discovered by NNPhD

Physics Example	Target	Ground Truth "New Physics"	NNPhD+Symbolic
Double Pendulum	$\begin{pmatrix}\ddot{\theta}_1\\ \ddot{\theta}_2\end{pmatrix}$	$\begin{pmatrix}\mathbf{-0.02}\dot{\theta}_1 - \mathbf{0.00}\dot{\theta}_2\\ \mathbf{-0.00}\dot{\theta}_1 - \mathbf{0.02}\dot{\theta}_2\end{pmatrix}$	$\begin{pmatrix}\mathbf{-0.018}\dot{\theta}_1 - \mathbf{0.001}\dot{\theta}_2\\ \mathbf{-0.001}\dot{\theta}_1 - \mathbf{0.018}\dot{\theta}_2\end{pmatrix}$
Neptune	$\begin{pmatrix}\ddot{x}\\ \ddot{y}\end{pmatrix}$	$\begin{pmatrix}\frac{\mathbf{0.005}(-x+\mathbf{3}\cos(\mathbf{0.192}t))}{[(x-\mathbf{3}\cos(\mathbf{0.192}t))^2+(y-\mathbf{3}\sin(\mathbf{0.192}t))^2]^{\frac{3}{2}}}\\ \frac{\mathbf{0.005}(-y+3\sin(\mathbf{0.192}t))}{[(y-\mathbf{3}\sin(\mathbf{0.192}t))^2+(y-\mathbf{3}\sin(\mathbf{0.192}t))^2]^{\frac{3}{2}}}\end{pmatrix}$	$\begin{pmatrix}\frac{\mathbf{0.0052}(-x+\mathbf{3.004}\cos(\mathbf{0.192}t))}{[(x-\mathbf{3.004}\cos(\mathbf{0.192}t))^2+(y-\mathbf{3.004}\sin(\mathbf{0.192}t))^2]^{\frac{3}{2}}}\\ \frac{\mathbf{0.0052}(-y+\mathbf{3.004}\sin(\mathbf{0.192}t))}{[(y-\mathbf{3.004}\sin(\mathbf{0.192}t))^2+(y-\mathbf{3.004}\sin(\mathbf{0.192}t))^2]^{\frac{3}{2}}}\end{pmatrix}$
Gravitational Radiation	$\begin{pmatrix}\ddot{x}_1\\ \ddot{y}_1\end{pmatrix}$	$\begin{pmatrix}\mathbf{-0.00165}(\dot{x}_1^2+\dot{y}_1^2)^{\mathbf{4}}\dot{x}_1\\ \mathbf{-0.00165}(\dot{x}_1^2+\dot{y}_1^2)^{\mathbf{4}}\dot{y}_1\end{pmatrix}$	$\begin{pmatrix}\mathbf{-0.00170}(\dot{x}_1^2+\dot{y}_1^2)^{\mathbf{3.94}}\dot{x}_1\\ \mathbf{-0.00170}(\dot{x}_1^2+\dot{y}_1^2)^{\mathbf{3.94}}\dot{y}_1\end{pmatrix}$

steps.

2. **Explaining:** After training, we aim to extract more interpretable physics from the UAN via constrained nonlinear optimization of free parameters (displayed as **bold** in Table II) to explain the output of the black-box, since ground truth symbolic forms are available.

In Table II, we show ground truth "new physics" and NNPhD discovered symbolic expressions. Fitted coefficients are seen to match ground truth quite well: (1) damping coefficient; (2) orbital radius and angular velocity of Neptune around the Sun; (3) magnitude and velocity dependence of gravitational wave emission.

B. Prediction of Trajectories

In addition to discovering new physics, as we saw above, NNPhD can also compete with other methods on simple trajectory prediction, and we will now test its performance for out-of-distribution generalization. Specifically, we test how accurately it can extrapolate the trajectory of the damped double pendulum from Section III A 2, whose state is specified by two angles (θ_1, θ_2) and corresponding angular velocities $(\dot{\theta}_1, \dot{\theta}_2)$. We compute a trajectory with a 4^{th}-order Runge-Kutta integrator at $N_{step} = 2000$ timesteps of size $\varepsilon = 0.1$ using the initial conditions $(\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2) = (1, 0, 0, 0)$. Our test task is to extrapolate beyond $t = 30$, so we split the trajectory into a training dataset $(0 \le t \le 30)$ and a test dataset $(30 \le t \le 200)$.

We train NNPhD with $\lambda = 0.2$ and feed its prediction f into a 4^{th}-order Runge-Kutta integrator to produce the predicted trajectory. Figure 4 compares the performance with that from a LNN and a pure black box neural network. The left panel shows that both NNPhD and the black box can fit θ_1 well on training samples and extrapolate for a short period, but fail at larger times due to accumulated errors and sensitive phases. In contrast, we see that the LNN cannot even fit the training data, because it has the invalid energy-conservation assumption built in. The right panel shows that ground-truth energy is decaying exponential over time due to friction, while the LNN stubbornly predicts constant energy. NNPhD is seen to predict the energy decay best of the three methods, while the black-box slightly overpredicts the the energy for a while and then incorrectly transitions to predicting approximate energy conservation.

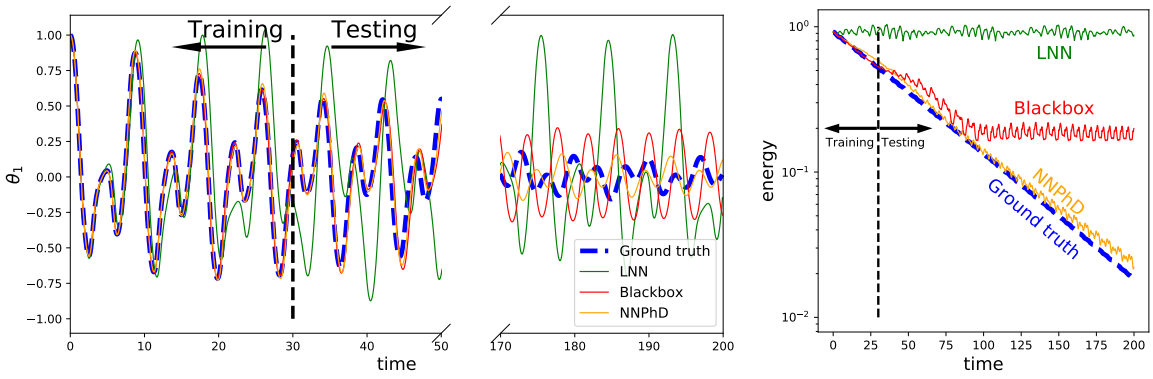
C. Theory Verification and Algorithm Benchmarking

In this section, to better understand its algorithmic behavior, we test NNPhD on the six simple dynamical systems in physics in Table III: conservative examples involve a harmonic oscillator (HO), a magnetic field (MF)[3] and constant gravity (CG) and non-conservative examples include linear damping (LD), constant damping (CD) and a periodic force (PF). We combine these into five examples to obtain two conservative systems (HO+MF, HO+CG) and three non-conservative systems (HO+LD, HO+CD, HO+PF), whose dynamical equations are summarized in Appendix A. For each system, we train NNPhD with the ADAM optimizer for 2,000 iterations, using batch size 32, learning rate schedule $\{0.01, 0.001, 0.0001, 0.00001\}$ and 500 iterations for each learning rate.

We now explore how the performance of the NNPhD depends on the regularization coefficient λ and norm index p by testing $\lambda = \{0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1, 2, 5, 10, 20, 50, 100\}$ and $p = 1, 2, 3$. Instead of simulating trajectories to generate data as in previous sections, we compute $\dot{\mathbf{q}} = f(\mathbf{q}, \dot{\mathbf{q}}, t)$ at N random points $(\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)}, t_i)$. We first generate all positions, velocities and times as independent Gaussian random variables with zero mean and unit standard deviation, then explore more complicated coverage in Section III D. We generate 10^3 training samples and 10^3 testing samples $(\mathbf{q}, \dot{\mathbf{q}}, t)$.

How performance depends on p: In Figure 5(a), we plot the dependence of the prediction error L_e on λ $(p = 1)$, again verifying the phase transition prediction from Section II E: The non-conservative systems (HO+LD, HO+CD, HO+PF) are seen to have a large error jump at $\lambda = 1$ while, in contrast, L_e does not increase at $\lambda = 1$ for the conservative systems (HO+MF, HO+CG). In fact, HO+MF has even lower prediction

[3] Note that we refer to the magnetic force as conservative because it conserves energy, even though physicists customarily limit that term to velocity-independent forces that can be written as the gradient of a potential function.



(a) (b)

FIG. 4: Double Pendulum example: (a) Both NNPhD and the black-box can accurately fit the angle θ_1 of training samples, and can successfully extrapolate for a brief period, while LNN fails to model the non-conservative dynamics; (b) NNPhD correctly predicts the exponential energy decay on testing samples, while the black-box generalizes worse, and LNN incorrectly conserves energy.

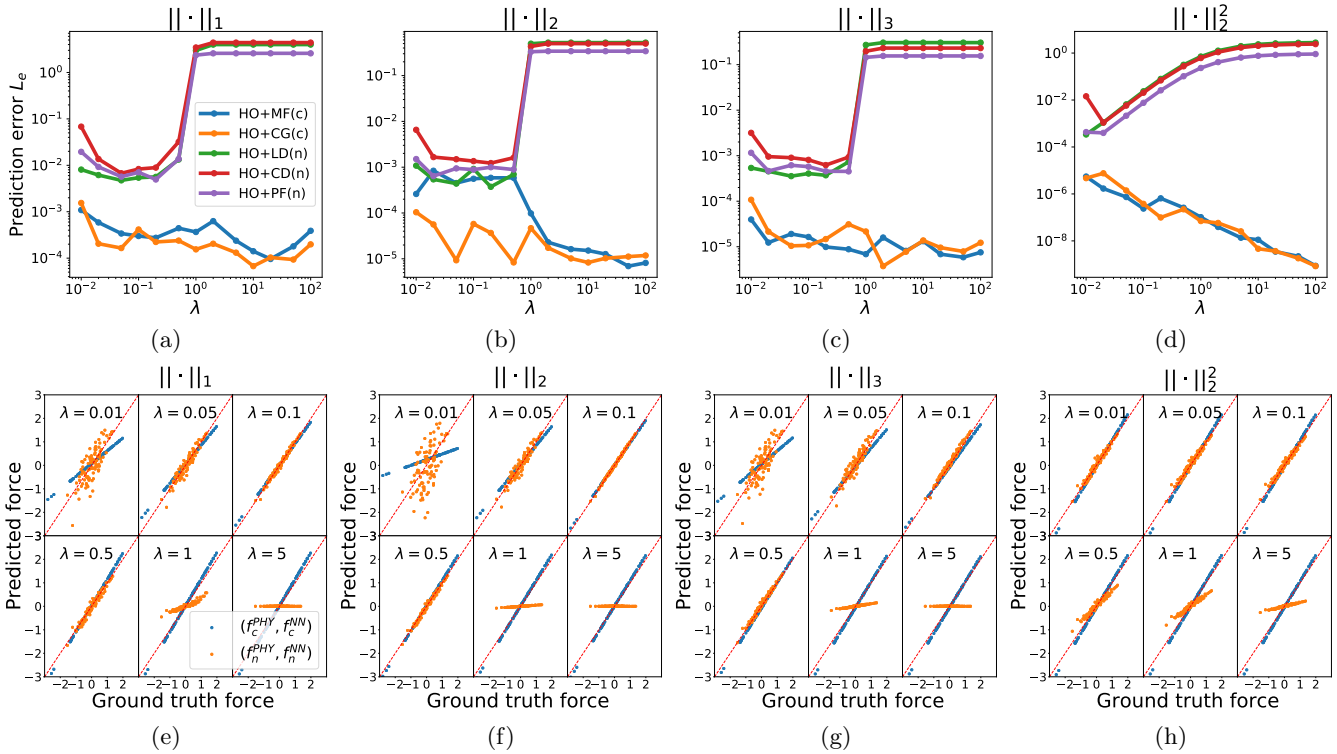


FIG. 5: NNPhD is seen to behave robustly for $0.05 \lesssim \lambda < 1$ and $p \geq 1$. We test NNPhD on five examples (the first two are conservative, and last three are non-conservative). (a)-(d) prediction error L_e as a function of λ with different norms as loss function: for (a)-(c) $\|\cdot\|_p (p = 1, 2, 3)$, non-conservative dynamics has an error jump at $\lambda = 1$, while conservative dynamics does not. In (d), mean squared loss leads to a smooth phase transition for non-conservative dynamics; (e)-(h) for the linear damping case $\ddot{q} = -q - \frac{1}{2}\dot{q}$, we show how f_c^{NN} and f_n^{NN} are aligned with f_c^{PHY} and f_n^{PHY} for different loss functions and different λ.

error at larger λ, showing the advantage of employing a Lagrangian Neural Network as opposed to a black box for conservative systems. Figure 5(a) shows that NNPhD has the ability to distinguish between conservative and non-conservative dynamics by looking at prediction loss around $\lambda = 1$, *i.e.*, a sharp phase transition indicates non-conservative dynamics. The above observations also apply to Figure 5(b)(c) when $p = 2$ and $p = 3$. However Figure 5(d) shows that mean-squared-error loss (where the L_2-norm is squared) leads to a smooth transition, known as second-order phase transition in physics.

How performance depends on λ: We then quantify how accurately the conservative and non-conservative components are modeled for different λ-values. Figure

TABLE III: Examples of Conservative and Non-conservative Dynamics

Classes	Model	Equation	Lagrangian
Conservative ($f_{\rm c}^{PHY}$)	Harmonic Oscillator (HO)	$\ddot{q} = -q$	$\mathcal{L} = \dot{q}^2/2 - q^2/2$
	Magnetic Field (MF)	$\ddot{q}_1 = \dot{q}_2$ $\ddot{q}_2 = -\dot{q}_1$	$\mathcal{L} = (\dot{q}_1 - q_2)^2/2$ $+(\dot{q}_2 + q_1)^2/2$
	Constant Gravity (CG)	$\ddot{q} = -1$	$\mathcal{L} = \dot{q}^2/2 - q$
Non-Conservative ($f_{\rm n}^{PHY}$)	Linear Damping (LD)	$\ddot{q} = -\dot{q}$	NA
	Constant Damping (CD)	$\ddot{q} = -\text{sgn}(\dot{q})$	
	Periodic Force (PF)	$\ddot{q} = \sin(t)$	

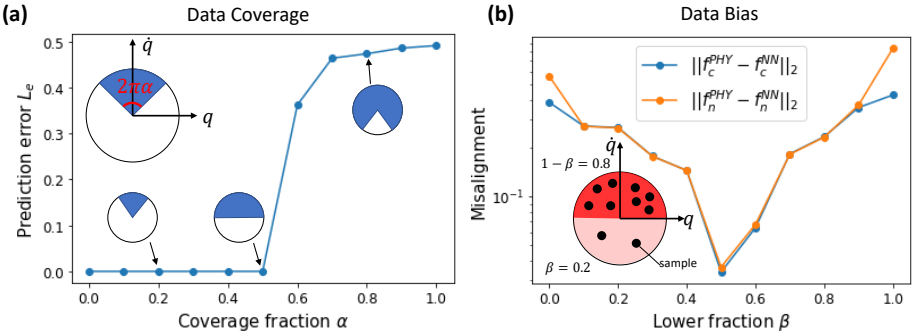


FIG. 6: Dependence on data distribution parameters α and β. Low quality data might prevent new physics discovery via (a) incomplete data coverage and (b) biased data distribution.

5(e) shows our results for the damped oscillator example $\ddot{q} = -q - \frac{1}{2}\dot{q}$, comparing $f_{\rm c}^{NN}$ with $f_{\rm c}^{PHY} = -q$ and $f_{\rm n}^{NN}$ with $f_{\rm n}^{PHY} = -\frac{1}{2}\dot{q}$. As Theorem 1 suggests, we observe that (1) when $\lambda > 1$, $f_{\rm n}^{NN}$ predicts 0 while $f_{\rm c}^{NN} \approx f_{\rm c}^{PHY}$; (2) when $0.05 \lesssim \lambda < 1$, $f_{\rm c}^{NN} \approx f_{\rm c}^{PHY}$ and $f_{\rm n}^{NN} \approx f_{\rm n}^{PHY}$; (3) when $\lambda \lesssim 0.05$, although in theory it behaves similarly to (2), a small λ does not have much incentive to penalize the black box, which therefore absorbs part of the conservative component. Figure 5(f)(g)(h) show that the alignments between the ground truth components and the the predictions from NNPhD are quite robust for different choices of loss function.

D. Physics Discovery Requires High-Quality Data

Although NNPhD does not assume any data distribution to achieve the decomposition, we will now see that NNPhD can only learn to accurately decompose the force into conservative and non-conservative parts if the data has high quality, specifically, if the data distribution has (1) adequate coverage of the state space $\mathbf{x} = (\mathbf{q}, \dot{\mathbf{q}})$ and (2) is unbiased.

Incomplete data coverage: We now explore the situation where data points are only accessible in a pie-shaped subset of space covering an angular fraction of $\alpha \in [0, 1]$, as illustrated in Figure 6(a). We consider the 1D constant damped oscillator $\ddot{q} = -q - \frac{1}{2}\text{sgn}(\dot{q})$, train NNPhD with $\lambda = 10$ on datasets with different fractions α and calculate the prediction loss L_e. Recall that when $\lambda = 10$, a high prediction error L_e is a sign of non-conservation. Figure 6(a) shows that when $\alpha \leq 0.5$, no samples are generated in the lower half plane (where $\dot{q} < 0$), then the prediction error L_e is nearly zero, revealing no sign of non-conservation. For $\alpha > 0.5$, on the other hand, NNPhD has a large L_e, revealing the non-conservative nature of the damping force. This observation makes physical sense since, if only $\dot{q} > 0$ samples are observed, the damping force acts as a constant conservative force (like gravity) which can be included as a $(-\frac{1}{2}q)$ term in a Lagrangian $\mathcal{L} = \frac{1}{2}\dot{q}^2 - \frac{1}{2}q^2 - \frac{1}{2}q$, making the dynamics appear energy conserving.

Imbalanced data distribution Even in the case when data is available everywhere in all relevant parts of phase space, the data set can still be imbalanced, *e.g.*, contain more $\dot{q} > 0$ samples than $\dot{q} < 0$ ones. Figure 6(a) show that this is not a sever problem in the sense that it does not preclude us from identifying the *existence* of non-conservative dynamics, since the presence of since merely a few samples with $\dot{q} < 0$ suffices to give a clear signal of non-conservation. However, such imbalance may harm the accuracy of our decomposition. We consider the linear damped oscillator $\ddot{q} = -q - \frac{1}{2}\dot{q}$ where a fraction β of the data is in the upper half plane while the remaining fraction $1-\beta$ is in the lower half-plane. We set $\lambda = 0.5$, train on datasets with different β and compare learned conservative and non-conservative force fields with ground truth. We found the learned functions $f_{\rm c}^{NN}$ and $f_{\rm n}^{NN}$ are not necessarily aligned with the ground truth decomposition $f_{\rm c}^{PHY} = -q$

and $f_{\rm n}^{PHY} = -\frac{1}{2}\dot{q}$. We define misalignment as

$$m_{\rm c} = (\frac{1}{nN}\sum_{i=1}^{N} ||f_{\rm c}^{NN}(\mathbf{x}^{(i)}) - f_{\rm c}^{PHY}(\mathbf{x}^{(i)})||^2)^{\frac{1}{2}},$$
$$m_{\rm n} = (\frac{1}{nN}\sum_{i=1}^{N} ||f_{\rm n}^{NN}(\mathbf{x}^{(i)}, t^{(i)}) - f_{\rm n}^{PHY}(\mathbf{x}^{(i)}, t^{(i)})||^2)^{\frac{1}{2}}. \tag{12}$$

Figure 6(b) shows this misalignment as a function of β, revealing a minimum with nearly zero misalignment for the $\beta = 0.5$ case when the data is balanced. In summary, these last numerical experiments show that high-quality data is important for new physics discovery, regardless of whether the data is analyzed by intelligent human scientists or machine learning.

IV. CONCLUSION

We have presented the Neural New-physics Detector (NNPhD), a method for decomposing a general force field into components that do and do not conserve energy. We showed that NNPhD was able to do this robustly for a series of physical examples without access to symbolic equations, providing clear evidence of the existence of conservation-violating new physics. We also found that NNPhD could extrapolate time series more accurately than both LNN and black-box neural networks. As ever-larger science and engineering datasets become available for dynamical systems, we hope that NNPhD will help enable more accurate prediction as well as aid discovery of interesting new phenomena.

Acknowledgements We thank Yuanqi Du and Jieyu Zhang for helpful discussions, and the Center for Brains, Minds, and Machines (CBMM) for hospitality. This work was supported by the Casey and Family Foundation, the Foundational Questions Institute, the Rothberg Family Fund for Cognitive Science, and AI Institute for Artificial Intelligence and Fundamental Interactions (IAIFI) through NSF Grant No. PHY-2019786.

[1] C. Burgess and G. Moore, *The standard model: A primer* (Cambridge University Press, 2007).
[2] I. M. Hutchings, Leonardo da vinci studies of friction, Wear **360-361**, 51 (2016).
[3] Wikipedia contributors, Discovery of neptune — Wikipedia, the free encyclopedia, `https://en.wikipedia.org/w/index.php?title=Discovery_of_Neptune&oldid=1000734782` (2021), [Online; accessed 23-February-2021].
[4] Wikipedia contributors, Cowan–reines neutrino experiment — Wikipedia, the free encyclopedia, `https://en.wikipedia.org/w/index.php?title=Cowan%E2%80%93Reines_neutrino_experiment&oldid=1000625707` (2021), [Online; accessed 23-February-2021].
[5] M. S. Turner, The dark side of the universe: from zwicky to accelerated expansion, Physics Reports **333**, 619 (2000).
[6] V. C. Rubin, Dark matter in spiral galaxies, Scientific American **248**, 96 (1983).
[7] A. Wolszczan and D. A. Frail, A planetary system around the millisecond pulsar psr1257+ 12, Nature **355**, 145 (1992).
[8] L. Esposito and E. Harrison, Properties of the hulse-taylor binary pulsar system, The Astrophysical Journal **196**, L1 (1975).
[9] M. Cranmer, S. Greydanus, S. Hoyer, P. Battaglia, D. Spergel, and S. Ho, Lagrangian neural networks, arXiv preprint arXiv:2003.04630 (2020).
[10] S. Greydanus, M. Dzamba, and J. Yosinski, Hamiltonian neural networks, in *Advances in Neural Information Processing Systems* (2019) pp. 15379–15389.
[11] M. Finzi, K. A. Wang, and A. G. Wilson, Simplifying hamiltonian and lagrangian neural networks via explicit constraints, Advances in Neural Information Processing Systems **33** (2020).
[12] A. Choudhary, J. F. Lindner, E. G. Holliday, S. T. Miller, S. Sinha, and W. L. Ditto, Forecasting hamiltonian dynamics without canonical coordinates, arXiv preprint arXiv:2010.15201 (2020).
[13] P. Jin, Z. Zhang, A. Zhu, Y. Tang, and G. E. Karniadakis, Sympnets: Intrinsic structure-preserving symplectic networks for identifying hamiltonian systems, Neural Networks **132**, 166 (2020).
[14] P. Toth, D. Jimenez Rezende, A. Jaegle, S. Racanière, A. Botev, and I. Higgins, Hamiltonian Generative Networks, arXiv e-prints , arXiv:1909.13789 (2019), arXiv:1909.13789 [cs.LG].
[15] Z. Long, Y. Lu, X. Ma, and B. Dong, Pde-net: Learning pdes from data, in *International Conference on Machine Learning* (PMLR, 2018) pp. 3208–3216.
[16] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud, Neural ordinary differential equations, Advances in neural information processing systems **31**, 6571 (2018).
[17] P. W. Battaglia, R. Pascanu, M. Lai, D. Rezende, and K. Kavukcuoglu, Interaction networks for learning about objects, relations and physics, arXiv preprint arXiv:1612.00222 (2016).
[18] F. Alet, E. Weng, T. Lozano-Pérez, and L. P. Kaelbling, Neural relational inference with fast modular meta-learning, in *Advances in Neural Information Processing Systems*, Vol. 32, edited by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Curran Associates, Inc., 2019) pp. 11827–11838.
[19] P. Y. Lu, S. Kim, and M. Soljačić, Extracting interpretable physical parameters from spatiotemporal systems using unsupervised learning, Phys. Rev. X **10**, 031056 (2020).
[20] K. Champion, B. Lusch, J. N. Kutz, and S. L. Brunton, Data-driven discovery of coordinates and governing equations, Proceedings of the National Academy of Sciences **116**, 22445 (2019).

[21] S.-M. Udrescu and M. Tegmark, Symbolic pregression: Discovering physical laws from raw distorted video, arXiv preprint arXiv:2005.11212 (2020).
[22] S. Kim, P. Lu, S. Mukherjee, M. Gilbert, L. Jing, V. Ceperic, and M. Soljacic, Integration of neural network-based symbolic regression in deep learning for scientific discovery, arXiv preprint arXiv:1912.04825 (2019).
[23] 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, Journal of Computational Physics **378**, 686 (2019).
[24] T. Matsubara, A. Ishikawa, and T. Yaguchi, Deep energy-based modeling of discrete-time physics, arXiv e-prints , arXiv (2019).
[25] M. Cranmer, A. Sanchez-Gonzalez, P. Battaglia, R. Xu, K. Cranmer, D. Spergel, and S. Ho, Discovering symbolic models from deep learning with inductive biases, arXiv preprint arXiv:2006.11287 (2020).
[26] M. Raissi, A. Yazdani, and G. E. Karniadakis, Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations, Science **367**, 1026 (2020).
[27] T. Wu and M. Tegmark, Toward an artificial intelligence physicist for unsupervised learning, Phys. Rev. E **100**, 033311 (2019).
[28] S.-H. Li, C.-X. Dong, L. Zhang, and L. Wang, Neural canonical transformation with symplectic flows, Phys. Rev. X **10**, 021020 (2020).
[29] M. Lutter, C. Ritter, and J. Peters, Deep lagrangian networks: Using physics as model prior for deep learning, arXiv preprint arXiv:1907.04490 (2019).
[30] Z. Liu and M. Tegmark, Machine learning conservation laws from trajectories, Phys. Rev. Lett. **126**, 180604 (2021).
[31] G. Welch, G. Bishop, *et al.*, An introduction to the kalman filter (1995).
[32] V. L. Guen, Y. Yin, J. Dona, I. Ayed, E. de Bézenac, N. Thome, and P. Gallinari, Augmenting physical models with deep networks for complex dynamics forecasting, arXiv preprint arXiv:2010.04456 (2020).
[33] A. Ajay, J. Wu, N. Fazeli, M. Bauza, L. P. Kaelbling, J. B. Tenenbaum, and A. Rodriguez, Augmenting physical simulators with stochastic neural networks: Case study of planar pushing and bouncing, in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2018) pp. 3066–3073.
[34] J. Wu, I. Yildirim, J. J. Lim, B. Freeman, and J. Tenenbaum, Galileo: Perceiving physical object properties by integrating a physics engine with deep learning, Advances in neural information processing systems **28**, 127 (2015).
[35] Y. D. Zhong, B. Dey, and A. Chakraborty, Dissipative symoden: Encoding hamiltonian dynamics with dissipation and control into deep learning, arXiv preprint arXiv:2002.08860 (2020).
[36] L. Scientific, V. collaborations, B. Abbott, R. Abbott, T. Abbott, M. Abernathy, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, *et al.*, The basic physics of the binary black hole merger gw150914, Annalen der Physik **529**, 1600209 (2017).
[37] J. Hanc, S. Tuleja, and M. Hancova, Symmetries and conservation laws: Consequences of noether's theorem, American Journal of Physics **72**, 428 (2004).
[38] H. Goldstein, C. Poole, and J. Safko, Classical mechanics (2002).

Appendix A: Toy Example details

For each dynamical system, the left hand side is $\ddot{\mathbf{q}}$, and right hand side is physical ground truth where conservative and non-conservative dynamics is explicitly separated as $\{f_{\mathrm{c}}^{PHY}(\mathbf{q}, \dot{\mathbf{q}})\} + \{f_{\mathrm{n}}^{PHY}(\mathbf{q}, \dot{\mathbf{q}}, t)\}$.

HO+MF ($k = B = 1$):

$$\begin{pmatrix} \ddot{x} \\ \ddot{y} \end{pmatrix} = \{-k \begin{pmatrix} x \\ y \end{pmatrix} + B \begin{pmatrix} \dot{y} \\ -\dot{x} \end{pmatrix}\} + \{\begin{pmatrix} 0 \\ 0 \end{pmatrix}\} \tag{A1}$$

HO+CG ($k = g = 1$)

$$\ddot{x} = \{-kx - g\} + \{0\} \tag{A2}$$

HO+LD ($k = 1, \gamma = \frac{1}{2}$)

$$\ddot{x} = \{-kx\} + \{-\gamma \dot{x}\} \tag{A3}$$

HO+CD ($k = 1, \gamma = \frac{1}{2}$)

$$\ddot{x} = \{-kx\} + \{-\gamma \mathrm{sgn}(\dot{x})\} \tag{A4}$$

HO+PF ($k = 1, a = \frac{1}{2}$)

$$\ddot{x} = \{-kx\} + \{a \sin(t)\} \tag{A5}$$

Appendix B: Neural network training details

We parameterize both the our LNN (conservative) and our UAN (non-conservative) force models as non-weight-sharing fully connected feedforward neural networks with two hidden 200-neuron layers. The LNN uses a mixture of softplus and quadratic activation (see Appendix C for details) and has Eq. (3) hard-coded right before outputting f_{c}^{NN}, while the UAN uses LeakyReLU activation (with negative slope $\alpha = 0.2$) and does not involve in any other inductive biases.

We measure the performance of NNPhD for

$$\lambda \in \{.01, .02, .05, .1, .2, .5, 1, 2, 5, 10, 20, 50, 100\} \tag{B1}$$

by first initializing and training NNPhD with $\lambda = 0.01$ using the ADAM optimizer with learning rate $\{10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}\}$ for 2000 steps (500 steps for each learning rate), and iteratively increasing λ and train for 2000 steps for each new λ-value. The model parameters of LNN and UAN are updated simultaneously.

Appendix C: Tricks to Boost LNN Training

As mentioned in [9], LNN is unstable and inefficient to train with traditional initializations in ML. As a result, expensive grid search of proper initializations is required. We propose two simpler tricks that have some improvements and are easy to implement. We use the example of a harmonic oscillator. The Lagrangian $\mathcal{L} = \frac{1}{2}\dot{q}^2 - \frac{1}{2}q^2$ contains only quadratic terms. We build a two hidden-layer networks with width 4-200-200-2.

Activation Trick: [9] uses Softplus as activation function, which is general but inefficient to represent a quadratic function. However the quadratic function is common and useful in physics, we propose to divide neurons into two groups, where one group uses Softplus as activation, and the other group uses quadractic function as activation.

Split Trick: One of instability when forwarding LNN comes from inversion of $\nabla_{\dot{\mathbf{q}}}\nabla_{\dot{\mathbf{q}}}\mathcal{L}$. In physical terms, $\nabla_{\dot{\mathbf{q}}}\nabla_{\dot{\mathbf{q}}}\mathcal{L}$ represents a mass scalar (matrix) which is positive (positive definite). However this constraint is not explicitly embedded to LNN, leading to training instabilities. We split $\mathcal{L}$ into two parts:

$$\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2 = \mathcal{L}_{NN} + \frac{1}{2} a \dot{\mathbf{q}}^T \dot{\mathbf{q}} \tag{C1}$$

where $\mathcal{L}_1$ is learned by LNN, while $\mathcal{L}_2$ is a fixed quadratic term (we choose $a = 1$). At initializations when $\mathcal{L}_{NN} \approx 0$, $\mathcal{L} \approx \frac{1}{2} a \dot{\mathbf{q}}^T \dot{\mathbf{q}}$ is positive definite.

To test how the proposed two tricks operate, we implement four models in Figure 7 to fit 1D harmonic oscillator: Softplus or quadratic activation, and w/wo the split trick. The best performance one is the LNN using quadratic activation and the split trick.

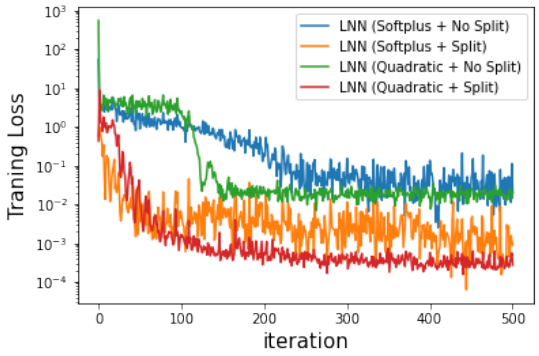


FIG. 7: Tricks to boost LNN training

Appendix D: Lagrangian mechanics for machine learning readers

For readers whose background is primarily in machine learning rather than physics, this section provides a brief review of the Lagrangian mechanics formalism that we use in this paper.

Conservative dynamics describes a dynamic where there exist conserved quantities (energy, momentum, angular momentum etc). Conservation laws are important in physics, as it corresponds to symmetries of our mother nature, according to Noether's theorem [37]. In particular, energy conservation is equivalent to time translational symmetry. To describe dynamics that conserves energy, physicists employ (time-independent) *Lagrangian* or *Hamiltonian* formulation. Since our work and prior work Lagrangian Neural Network (LNN) [9] are based on Lagrangian mechanics, we provide a brief introduction here.

The Lagrangian formalism models a classical physics system with trajectory $\mathbf{x}(t) = (\mathbf{q}, \dot{\mathbf{q}})$ that begins in one state $\mathbf{x}(t_0)$ and ends up in another state $\mathbf{x}(t_1)(t_1 > t_0)$, where $\mathbf{q}$ and $\dot{\mathbf{q}}$ are called the generalized coordinates and velocities respectively. There are many paths that these states might take as they pass from $\mathbf{x}(t_0)$ to $\mathbf{x}(t_1)$, and Lagrangian mechanics tells that there is only one path that the physical system will take, i.e., the path that minimizes $\int_{t_0}^{t_1}(T(\mathbf{q}(t), \dot{\mathbf{q}}(t)) - V(\mathbf{q}(t), \dot{\mathbf{q}}(t)))dt$, where T is kinetic energy and V is the potential energy. The term $\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}) \equiv T(\mathbf{q}, \dot{\mathbf{q}}) - V(\mathbf{q}, \dot{\mathbf{q}})$ is called Lagrangian and the path (trajectory) of the system is determined by *Euler-Lagrange equation*:

$$\frac{d}{dt}\nabla_{\dot{\mathbf{q}}}\mathcal{L} = \nabla_{\mathbf{q}}\mathcal{L}.$$

Based on the formulas in Lagrangian Neural Network (LNN) [9], Euler-Lagrange equation $\frac{d}{dt}\nabla_{\dot{\mathbf{q}}}\mathcal{L} = \nabla_{\mathbf{q}}\mathcal{L}$ can be rewritten by applying a chain rule $\frac{d}{dt}\nabla_{\dot{\mathbf{q}}}^T\mathcal{L} = (\nabla_{\dot{\mathbf{q}}}\nabla_{\dot{\mathbf{q}}}^T\mathcal{L})\ddot{\mathbf{q}} + (\nabla_{\mathbf{q}}\nabla_{\dot{\mathbf{q}}}\mathcal{L})\dot{\mathbf{q}}$ resulting in:

$$\ddot{\mathbf{q}} = (\nabla_{\dot{\mathbf{q}}}\nabla_{\dot{\mathbf{q}}}^T\mathcal{L})^{-1}(\nabla_{\mathbf{q}}\mathcal{L} - (\nabla_{\mathbf{q}}\nabla_{\dot{\mathbf{q}}}^T\mathcal{L})\dot{\mathbf{q}}) \tag{D1}$$

One inductive bias brought by Lagrangian mechanics is that Eq. (3) describes **conservative physical dynamics**. That is, the energy function defined as

$$H(\mathbf{q}, \dot{\mathbf{q}}) = \nabla_{\dot{\mathbf{q}}}^T\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - \mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}) \tag{D2}$$

is constant along a trajectory $(\mathbf{q}(t), \dot{\mathbf{q}}(t))$ driven by Eq. (3). The proof of $H(\mathbf{q}, \dot{\mathbf{q}})$ conservation can be found in standard physics textbooks [38] and is included here for completeness.

Lemma 1. *Given a Lagrangian $\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})$, the energy defined in Eq. (3) is conserved along the trajectory $(\mathbf{q}(t), \dot{\mathbf{q}}(t))$ driven by Eq. (2).*

Proof. Invoke the chain rule one obtains the time derivative of $\mathcal{L}$:

$$\frac{d\mathcal{L}}{dt} = \dot{\mathbf{q}}^T\nabla_{\mathbf{q}}\mathcal{L} + \ddot{\mathbf{q}}^T\nabla_{\dot{\mathbf{q}}}\mathcal{L} \tag{D3}$$

Eq. (4) is equivalent to Euler-Lagrangian equation $\frac{d}{dt}\nabla_{\dot{\mathbf{q}}}\mathcal{L} = \nabla_{\mathbf{q}}\mathcal{L}$, so we replace $\nabla_{\mathbf{q}}\mathcal{L}$ with $\frac{d}{dt}\nabla_{\dot{\mathbf{q}}}\mathcal{L}$:

$$\frac{d\mathcal{L}}{dt} = \dot{\mathbf{q}}^T\frac{d}{dt}\nabla_{\dot{\mathbf{q}}}\mathcal{L} + \ddot{\mathbf{q}}^T\nabla_{\dot{\mathbf{q}}}\mathcal{L} = \frac{d}{dt}(\dot{\mathbf{q}}^T\nabla_{\dot{\mathbf{q}}}\mathcal{L}) \longrightarrow \frac{dH}{dt} \equiv \frac{d}{dt}(\dot{\mathbf{q}}^T\nabla_{\dot{\mathbf{q}}}\mathcal{L} - \mathcal{L}) = 0 \tag{D4}$$

□

Since not all physical system conserves energy, Lagrangian mechanics is insufficient to describe non-conservative dynamics, motivating the design of NNPhD framework. We prove that linear damp example is non-conservative, i.e., it cannot be represented by Lagrangian mechanics.

Lemma 2. *Let function $f : \mathbb{R}^2 \to \mathbb{R}$ be defined as $f(\mathbf{q}, \dot{\mathbf{q}}) = c\dot{\mathbf{q}}$, where c can be any real non-zero constant. Then, f cannot be represented by Eq. (3) for any Lagrangian $\mathcal{L} \in D^2(\mathbf{q}, \dot{\mathbf{q}})$ ($D^2(\mathbf{q}, \dot{\mathbf{q}})$ is the function space consisting of all twice-differentiable functions with respect to $(\mathbf{q}, \dot{\mathbf{q}})$).*

Proof. We prove the claim by reduction to absurdity. Suppose there exists a Lagrangian $\mathcal{L} \in D^2(\mathbf{q}, \dot{\mathbf{q}})$, such that,

$$c\dot{\mathbf{q}} = \left(\frac{\partial^2 \mathcal{L}}{\partial \dot{\mathbf{q}}^2}(\mathbf{q}, \dot{\mathbf{q}})\right)^{-1} \left(\frac{\partial \mathcal{L}}{\partial \mathbf{q}}(\mathbf{q}, \dot{\mathbf{q}}) - \left(\frac{\partial^2 \mathcal{L}}{\partial \dot{\mathbf{q}} \partial \mathbf{q}}(\mathbf{q}, \dot{\mathbf{q}})\right) \dot{\mathbf{q}}\right). \tag{D5}$$

By multiplying $\left(\frac{\partial^2 \mathcal{L}}{\partial \dot{\mathbf{q}}^2}(\mathbf{q}, \dot{\mathbf{q}})\right)$ to both sides of eq. (D5), we have

$$c\left(\frac{\partial^2 \mathcal{L}}{\partial \dot{\mathbf{q}}^2}(\mathbf{q}, \dot{\mathbf{q}})\right) \dot{\mathbf{q}} = \frac{\partial \mathcal{L}}{\partial \mathbf{q}}(\mathbf{q}, \dot{\mathbf{q}}) - \left(\frac{\partial^2 \mathcal{L}}{\partial \dot{\mathbf{q}} \partial \mathbf{q}}(\mathbf{q}, \dot{\mathbf{q}})\right) \dot{\mathbf{q}},$$

which by eq. (D2) further leads to

$$c\frac{\partial H}{\partial \dot{\mathbf{q}}}(\mathbf{q}, \dot{\mathbf{q}}) + \frac{\partial H}{\partial \mathbf{q}}(\mathbf{q}, \dot{\mathbf{q}}) = 0. \tag{D6}$$

By variable substitution, let $H(\mathbf{q}, \dot{\mathbf{q}}) = g(c\mathbf{q} + \dot{\mathbf{q}}, \dot{\mathbf{q}} - c\mathbf{q})$. By eq. (D6),

$$\begin{aligned}\frac{\partial g(c\mathbf{q} + \dot{\mathbf{q}}, \dot{\mathbf{q}} - c\mathbf{q})}{\partial (c\mathbf{q} + \dot{\mathbf{q}})} =& \frac{\partial H(\mathbf{q}, \dot{\mathbf{q}})}{\partial \mathbf{q}} \frac{\partial \mathbf{q}}{\partial (c\mathbf{q} + \dot{\mathbf{q}})} + \frac{\partial H(\mathbf{q}, \dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}} \frac{\partial \dot{\mathbf{q}}}{\partial (c\mathbf{q} + \dot{\mathbf{q}})} \\ =& \frac{1}{2c} \frac{\partial H(\mathbf{q}, \dot{\mathbf{q}})}{\partial \mathbf{q}} + \frac{1}{2} \frac{\partial H(\mathbf{q}, \dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}} = 0.\end{aligned}$$

Therefore, $H(\mathbf{q}, \dot{\mathbf{q}})$ is invariant of $c\mathbf{q} + \dot{\mathbf{q}}$ and only relies on the value of $\dot{\mathbf{q}} - c\mathbf{q}$. Thus, we can further abbreviate $H(\mathbf{q}, \dot{\mathbf{q}})$ as $g(\dot{\mathbf{q}} - c\mathbf{q})$.
On the other hand, by eq. (D2),

$$g(-c\mathbf{q}) = g(0 - c\mathbf{q}) = H(\mathbf{q}, \dot{\mathbf{q}}) = -\mathcal{L}(\mathbf{q}, 0).$$

Therefore,

$$\begin{aligned}\left.\frac{\partial \mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}}\right|_{\dot{\mathbf{q}}=0} &= \lim_{\dot{\mathbf{q}} \to 0} \frac{\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}) - \mathcal{L}(\mathbf{q}, 0)}{\dot{\mathbf{q}}} \\ &= \lim_{\dot{\mathbf{q}} \to 0} \frac{\dot{\mathbf{q}} \frac{\partial \mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}}(\mathbf{q}, \dot{\mathbf{q}}) - g(\dot{\mathbf{q}} - c\mathbf{q}) + g(-c\mathbf{q})}{\dot{\mathbf{q}}} \\ &\overset{(*)}{=} \left.\frac{\partial \mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}}\right|_{\dot{\mathbf{q}}=0} - g'(-c\mathbf{q}),\end{aligned}$$

where eq. $(*)$ is due to that $\frac{\partial \mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}}(\mathbf{q}, \dot{\mathbf{q}})$ is differentiable (thus continuous).
Therefore, we have $g'(\mathbf{q}) = 0$ for any $\mathbf{q}$, which further leads to $H(\mathbf{q}, \dot{\mathbf{q}})$ is a constant function and

$$c\dot{\mathbf{q}} = -\left(\frac{\partial^2 \mathcal{L}}{\partial \dot{\mathbf{q}}^2}(\mathbf{q}, \dot{\mathbf{q}})\right)^{-1} \frac{\partial H(\mathbf{q}, \dot{\mathbf{q}})}{\partial \mathbf{q}} = 0.$$

The proof is completed since $c \neq 0$. □

Appendix E: Learning perspectives of Section II C

We describe the learning task based on the force decomposition in Section II C. Given samples $(\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)}, t^{(i)}; \ddot{\mathbf{q}}^{(i)}), i = 1, \cdots, N$ that are uniformly drawn from the trajectories of dynamic $\ddot{\mathbf{q}} = f(\mathbf{q}, \dot{\mathbf{q}}, t)$ with $t \in [0, T]$ or a given distribution μ, we aim to learn both f_{c} and f_{n} from data. Because the ground-truth dynamic and its vector space are unknown, we need to select a model space $(\mathcal{G}, \|\cdot\|_{\mathcal{G}})$ which is also a normed vector space to find the best model in it. For this learning problem, we learn the model pair $(f_{\mathrm{c}}^{NN}, f_{\mathrm{n}}^{NN})$ simultaneously by solving the following constrained minimization problem

$$\min_{(f_{\mathrm{n}}^{NN}, f_{\mathrm{c}}^{NN})} \mathcal{L}_S(f_{\mathrm{n}}^{NN}, f_{\mathrm{c}}^{NN}) = \frac{1}{N}\sum_{i=1}^{N} \|f_{\mathrm{c}}^{NN}(\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)}) + f_{\mathrm{n}}^{NN}(\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)}, t^{(i)}) - \ddot{\mathbf{q}}^{(i)}\|_{\mathcal{G}}$$

$$s.t. \quad f_{\mathrm{c}}^{NN} = \arg\min_{g \in \mathcal{G}_{\mathrm{c}}} \frac{1}{N}\sum_{i=1}^{N} \|\ddot{\mathbf{q}}^{(i)} - g(\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)})\|_{\mathcal{G}}$$

We make the following discussions on the learning task: We denote the optimal models of the above optimization problem as $(f_{\mathrm{c}}^{NN*}, f_{\mathrm{n}}^{NN*})$. The interpolating prediction ability (which is measured by the gap between $\int_0^T \|f_{\mathrm{c}}^{NN*} + f_{\mathrm{n}}^{NN*} - \ddot{\mathbf{q}}\| dt$ and $\mathcal{L}_S(f_{\mathrm{c}}^{NN*}, f_{\mathrm{n}}^{NN*})$) is determined by the approximation ability of $\mathcal{G}$ and the number of training data. As the number of training data N increases, the gap will be smaller. As the approximation ability of $\mathcal{G}$ becomes stronger, the gap will be smaller.

Appendix F: Theorem 1 (formal)

Theorem 1. *We suppose the ground-truth hypothesis space $(\mathcal{G}, \|\cdot\|_p)$, Let $f_{\mathrm{c}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}; \boldsymbol{w}_{\mathrm{c}})$ be the Lagrangian Neural Network with parameters $\boldsymbol{w}_{\mathrm{c}}$ in NNPhD framework, and $f_{\mathrm{n}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}, t; \boldsymbol{w}_{\mathrm{n}})$ be the black box neural network with parameters $\boldsymbol{w}_{\mathrm{n}}$ in NNPhD framework. Assume the black box neural network can interpolate every continuous function of $(\mathbf{q}, \dot{\mathbf{q}}, t)$ at any N points, i.e., for any dataset $\{(\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)}, t^{(i)})\}_{i=1}^N$ with distinguished elements, $\{\{f_{\mathrm{n}}^{NN}(\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)}, t^{(i)}; \boldsymbol{w}_{\mathrm{n}})\}_{i=1}^N : \boldsymbol{w}_{\mathrm{n}} \in \mathbb{R}^{d_{\boldsymbol{w}_{\mathrm{n}}}}\} = \mathcal{R}^N$, where $d_{\boldsymbol{w}_{\mathrm{n}}}$ is the dimension of $\boldsymbol{w}_{\mathrm{n}}$. Then, given any continuous function f and any norm $\|\cdot\|$ on function space $\mathcal{C}(\mathbf{q}, \dot{\mathbf{q}}, t)$, the following claim stands:*
(1) For $\lambda > 1$, optimizing L_{NNPhD} is equivalent to optimize $\|f_{\mathrm{c}}^{NN}(\cdot, \cdot; \boldsymbol{w}_{\mathrm{c}}) - \ddot{\mathbf{q}}\|$ while keeping $f_{\mathrm{n}}^{NN}(\cdot, \cdot, \cdot; \boldsymbol{w}_{\mathrm{n}})$ as zero, that is,

$$\arg\min_{\boldsymbol{w}_{\mathrm{c}}, \boldsymbol{w}_{\mathrm{n}}} \left(\frac{1}{N}\sum_{i=1}^{N} \|f_{\mathrm{c}}^{NN}(\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)}; \boldsymbol{w}_{\mathrm{c}}) + f_{\mathrm{n}}^{NN}(\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)}, t^{(i)}; \boldsymbol{w}_{\mathrm{n}}) - \ddot{\mathbf{q}}^{(i)}\|^p\right)^{\frac{1}{p}} + \lambda\left(\frac{1}{N}\|f_{\mathrm{n}}^{NN}(\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)}, t^{(i)}; \boldsymbol{w}_{\mathrm{n}})\|^p\right)^{\frac{1}{p}}$$

$$= \left\{(\boldsymbol{w}_{\mathrm{c}}, \boldsymbol{w}_{\mathrm{n}}) : \boldsymbol{w}_{\mathrm{c}} \in \arg\min_{\boldsymbol{w}_{\mathrm{c}}} \quad \left(\frac{1}{N}\sum_{i=1}^{N} \|f_{\mathrm{c}}^{NN}(\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)}; \boldsymbol{w}_{\mathrm{c}}) - \ddot{\mathbf{q}}^{(i)}\|^p\right)^{\frac{1}{p}}, \quad f_{\mathrm{n}}^{NN}(\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)}, t^{(i)}; \boldsymbol{w}_{\mathrm{n}}) = 0\right\}.$$

(2) For $0 < \lambda < 1$, optimizing L_{NNPhD} is also equivalent to optimize $\|f_{\mathrm{c}}^{NN}(\cdot, \cdot; \boldsymbol{w}_{\mathrm{c}}) - \ddot{\mathbf{q}}\|$ while keeping $f_{\mathrm{n}}^{NN}(\cdot, \cdot, \cdot; \boldsymbol{w}_{\mathrm{n}})$ as $f - f_{\mathrm{c}}^{NN}(\cdot, \cdot; \boldsymbol{w}_{\mathrm{c}})$, that is,

$$\arg\min_{\boldsymbol{w}_{\mathrm{c}}, \boldsymbol{w}_{\mathrm{n}}} \left(\frac{1}{N}\sum_{i=1}^{N} \|f_{\mathrm{c}}^{NN}(\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)}; \boldsymbol{w}_{\mathrm{c}}) + f_{\mathrm{n}}^{NN}(\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)}, t^{(i)}; \boldsymbol{w}_{\mathrm{n}}) - \ddot{\mathbf{q}}^{(i)}\|^p\right)^{\frac{1}{p}} + \lambda\left(\frac{1}{N}\|f_{\mathrm{n}}^{NN}(\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)}, t^{(i)}; \boldsymbol{w}_{\mathrm{n}})\|^p\right)^{\frac{1}{p}}$$

$$= \left\{(\boldsymbol{w}_{\mathrm{c}}, \boldsymbol{w}_{\mathrm{n}}) : \boldsymbol{w}_{\mathrm{c}} \in \arg\min_{\boldsymbol{w}_{\mathrm{c}}} \quad \left(\frac{1}{N}\sum_{i=1}^{N} \|f_{\mathrm{c}}^{NN}(\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)}; \boldsymbol{w}_{\mathrm{c}}) - \ddot{\mathbf{q}}^{(i)}\|^p\right)^{\frac{1}{p}}, \quad f_{\mathrm{n}}^{NN}(\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)}, t^{(i)}; \boldsymbol{w}_{\mathrm{n}}) = f(\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)}, t^{(i)}) - f_{\mathrm{c}}^{NN}(\mathbf{q}^{(i)}, \dot{\mathbf{q}}^{(i)}; \boldsymbol{w}_{\mathrm{c}})\right\}.$$

We will provide proof of Theorem 1. We will actually show our results hold for general norms which include the discrete norm we use in Theorem 1. Concretely, Theorem 1 holds as a special case as the following theorem:

Theorem 2 (Theorem 1, extended to general norms)**.** *Let $f_{\mathrm{c}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}; \boldsymbol{w}_{\mathrm{c}})$ be the Lagrangian Neural Network with parameters $\boldsymbol{w}_{\mathrm{c}}$ in NNPhD framework, and $f_{\mathrm{n}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}, t; \boldsymbol{w}_{\mathrm{n}})$ be the black box neural network with parameters $\boldsymbol{w}_{\mathrm{n}}$ in NNPhD framework. Assume the black box neural network can represent every continuous function of $(\mathbf{q}, \dot{\mathbf{q}}, t)$ under the norm $\|\cdot\|$, i.e., $\{g(\mathbf{q}, \dot{\mathbf{q}}, t) : \exists \boldsymbol{w}_{\mathrm{n}},\ \|g(\mathbf{q}, \dot{\mathbf{q}}, t) - f_{\mathrm{n}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}; \boldsymbol{w}_{\mathrm{n}})\| = 0\} = \mathcal{C}(\mathbf{q}, \dot{\mathbf{q}}, t)$. Then, given any continuous function f, the following claim stands:*
(1) For $\lambda > 1$,

$$\arg\min_{\boldsymbol{w}_{\mathrm{c}}, \boldsymbol{w}_{\mathrm{n}}} \|f(\mathbf{q}, \dot{\mathbf{q}}, t) - f_{\mathrm{c}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}; \boldsymbol{w}_{\mathrm{c}}) - f_{\mathrm{n}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}, t; \boldsymbol{w}_{\mathrm{n}})\| + \lambda\|f_{\mathrm{n}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}, t; \boldsymbol{w}_{\mathrm{n}})\|$$

$$= \left\{(\boldsymbol{w}_{\mathrm{c}}, \boldsymbol{w}_{\mathrm{n}}) : \boldsymbol{w}_{\mathrm{c}} \in \arg\min_{\boldsymbol{w}_{\mathrm{c}}} \|f(\mathbf{q}, \dot{\mathbf{q}}, t) - f_{\mathrm{c}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}; \boldsymbol{w}_{\mathrm{c}})\|, \quad \|f_{\mathrm{n}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}, t; \boldsymbol{w}_{\mathrm{n}})\| = 0\right\}.$$

(2) For $0 < \lambda < 1$,

$$\begin{aligned}&\arg\min_{\boldsymbol{w}_\mathrm{c},\boldsymbol{w}_\mathrm{n}} \|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c}) - f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\| + \lambda\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\| \\ =&\left\{(\boldsymbol{w}_\mathrm{c},\boldsymbol{w}_\mathrm{n}) : \boldsymbol{w}_\mathrm{c} \in \arg\min_{\boldsymbol{w}_\mathrm{c}} \|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})\|, \|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n}) - f(\mathbf{q},\dot{\mathbf{q}},t) + f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})\| = 0\right\}.\end{aligned}$$

Proof. We prove the two cases above separately.
(1) If $\lambda > 1$, for any $\boldsymbol{w}_\mathrm{c}$ and $\boldsymbol{w}_\mathrm{n}$,

$$\begin{aligned}&\|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c}) - f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\| + \lambda\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\| \\ =&\|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c}) - f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\| + \|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\| + (\lambda-1)\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\| \\ \overset{(*)}{\geq}&\|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})\| + (\lambda-1)\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|. \end{aligned} \tag{F1}$$

Since $\lambda - 1 > 0$,

$$\begin{aligned}&\arg\min_{\boldsymbol{w}_\mathrm{c},\boldsymbol{w}_\mathrm{n}} \left(\|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})\| + (\lambda-1)\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|\right) \\ =&\left(\arg\min_{\boldsymbol{w}_\mathrm{c}} \|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})\|, \arg_{\boldsymbol{w}_\mathrm{n}} \|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\| = 0\right).\end{aligned}$$

For any $(\boldsymbol{w}_\mathrm{c}^0, \boldsymbol{w}_\mathrm{n}^0)$ where $\boldsymbol{w}_\mathrm{c}^0 \in \arg\min_{\boldsymbol{w}_\mathrm{c}} \|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})\|$ and $\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{n}^0)\| = 0$, the equality of inequality $(*)$ of Eq. (F1) is obtained. Therefore,

$$(\boldsymbol{w}_\mathrm{c}^0, \boldsymbol{w}_\mathrm{n}^0) \in \arg\min_{\boldsymbol{w}_\mathrm{c},\boldsymbol{w}_\mathrm{n}} \|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c}) - f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\| + \lambda\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|,$$

which further leads to

$$\begin{aligned}&\arg\min_{\boldsymbol{w}_\mathrm{c},\boldsymbol{w}_\mathrm{n}} \left(\|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})\| + (\lambda-1)\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|\right) \\ \subset&\arg\min_{\boldsymbol{w}_\mathrm{c},\boldsymbol{w}_\mathrm{n}} \left(\|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c}) - f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\| + \lambda\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|\right),\end{aligned}$$

and

$$\begin{aligned}&\min_{\boldsymbol{w}_\mathrm{c},\boldsymbol{w}_\mathrm{n}} \left(\|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c}) - f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\| + \lambda\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|\right) \\ =&\min_{\boldsymbol{w}_\mathrm{c},\boldsymbol{w}_\mathrm{n}} \left(\|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})\| + (\lambda-1)\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|\right).\end{aligned} \tag{F2}$$

Combining Eq. (F1), Eq. (F2) further leads to

$$\begin{aligned}&\arg\min_{\boldsymbol{w}_\mathrm{c},\boldsymbol{w}_\mathrm{n}} \left(\|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c}) - f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\| + \lambda\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|\right) \\ \subset&\arg\min_{\boldsymbol{w}_\mathrm{c},\boldsymbol{w}_\mathrm{n}} \left(\|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})\| + (\lambda-1)\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|\right).\end{aligned}$$

The proof for $\lambda > 1$ is completed.
(2) If $\lambda < 1$, for any $\boldsymbol{w}_\mathrm{c}$ and $\boldsymbol{w}_\mathrm{n}$, we decompose $\|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c}) - f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\| + \lambda\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|$ as follows:

$$\begin{aligned}&\|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c}) - f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\| + \lambda\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\| \\ =&(\lambda + (1-\lambda))\|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c}) - f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\| + \lambda\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\| \\ \overset{(**)}{\geq}&\lambda\|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})\| + (1-\lambda)\|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c}) - f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|, \end{aligned} \tag{F3}$$

where eq. $(**)$ is due to triangle inequality.
On the other hand, for any fixed $\boldsymbol{w}_\mathrm{c}$, minimum of eq. (F3) is obtained if and only if $\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n}) - f(\mathbf{q},\dot{\mathbf{q}},t) + f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})\| = 0$, in which case equality of eq. $(**)$ is also obtained. Therefore, for a given $\boldsymbol{w}_\mathrm{c}$,

$$\begin{aligned}&\min_{\boldsymbol{w}_\mathrm{n}} \|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c}) - f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\| + \lambda\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\| \\ =&\lambda\|f(\mathbf{q},\dot{\mathbf{q}},t) - f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})\|,\end{aligned}$$

and

$$\begin{aligned}&\arg\min_{\boldsymbol{w}_\mathrm{n}} \|f(\mathbf{q},\dot{\mathbf{q}},t)-f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})-f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|+\lambda\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|\\ =&\{\boldsymbol{w}_\mathrm{n}:\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})-f(\mathbf{q},\dot{\mathbf{q}},t)+f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})\|=0\}.\end{aligned}\tag{F4}$$

Since

$$\begin{aligned}&\arg\min_{\boldsymbol{w}_\mathrm{c},\boldsymbol{w}_\mathrm{n}} \|f(\mathbf{q},\dot{\mathbf{q}},t)-f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})-f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|+\lambda\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|\\ =&\arg\min_{\boldsymbol{w}_\mathrm{c}}\min_{\boldsymbol{w}_\mathrm{n}} \|f(\mathbf{q},\dot{\mathbf{q}},t)-f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})-f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|+\lambda\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|,\end{aligned}$$

by applying eq. (F4), we finally have

$$\begin{aligned}&\arg\min_{\boldsymbol{w}_\mathrm{c},\boldsymbol{w}_\mathrm{n}} \|f(\mathbf{q},\dot{\mathbf{q}},t)-f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})-f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|+\lambda\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|\\ =&\left\{(\boldsymbol{w}_\mathrm{c},\boldsymbol{w}_\mathrm{n}):\boldsymbol{w}_\mathrm{c}\in\arg\min_{\boldsymbol{w}_\mathrm{c}}\lambda\|f(\mathbf{q},\dot{\mathbf{q}},t)-f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})\|,\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})-f(\mathbf{q},\dot{\mathbf{q}},t)+f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})\|=0\right\}\\ =&\left\{(\boldsymbol{w}_\mathrm{c},\boldsymbol{w}_\mathrm{n}):\boldsymbol{w}_\mathrm{c}\in\arg\min_{\boldsymbol{w}_\mathrm{c}}\|f(\mathbf{q},\dot{\mathbf{q}},t)-f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})\|,\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})-f(\mathbf{q},\dot{\mathbf{q}},t)+f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})\|=0\right\}.\end{aligned}$$

The proof is completed. □

The above theorem describes the case that $f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})$ can represent every continuous function. However, in practice, the black box neural network can only access functions close to the original solution. In this general case, we instead have

Corollary 2.1. *Let $f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})$ be the Lagrangian Neural Network with parameters $\boldsymbol{w}_\mathrm{c}$ in NNPhD framework, and $f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})$ be the black box neural network with parameters $\boldsymbol{w}_\mathrm{n}$ in NNPhD framework. Assume the black box neural network can approximate every continuous function of $t,\mathbf{q},\dot{\mathbf{q}}$ by error $\varepsilon>0$ under some norm $\|\cdot\|$ on function space $\mathcal{C}(\mathbf{q},\dot{\mathbf{q}},t)$, i.e., $\forall f\in\mathcal{C}(\mathbf{q},\dot{\mathbf{q}},t)$, there exists a $\boldsymbol{w}_\mathrm{n}^f$, such that, $\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t,\boldsymbol{w}_\mathrm{n}^f)-f(\mathbf{q},\dot{\mathbf{q}},t)\|\le\varepsilon$. Furthermore, assume there exists $\boldsymbol{w}_0$, such that, $f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_0)\equiv 0$. Then, given any continuous function f, the following claim stands:*

- *For $\lambda>1$,*

$$\begin{aligned}&\arg\min_{\boldsymbol{w}_\mathrm{c},\boldsymbol{w}_\mathrm{n}} \|f(\mathbf{q},\dot{\mathbf{q}},t)-f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})-f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|+\lambda\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|\\ =&\left(\arg\min_{\boldsymbol{w}_\mathrm{c}} \|f(\mathbf{q},\dot{\mathbf{q}},t)-f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})\|,0\right).\end{aligned}$$

- *For $\lambda<1$, for any $(\boldsymbol{w}_\mathrm{c}^0,\boldsymbol{w}_\mathrm{n}^0)\in\arg\min_{\boldsymbol{w}_\mathrm{c},\boldsymbol{w}_\mathrm{n}} \|f(\mathbf{q},\dot{\mathbf{q}},t)-f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})-f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|+\lambda\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|$,*

$$\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n}^0)-f(\mathbf{q},\dot{\mathbf{q}},t)+f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c}^0)\|\le\frac{1+\lambda}{1-\lambda}\varepsilon,\tag{F5}$$

$$\|f(\mathbf{q},\dot{\mathbf{q}},t)-f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c}^0)\|\le\min_{\boldsymbol{w}_\mathrm{c}}\|f(\mathbf{q},\dot{\mathbf{q}},t)-f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})\|+(1+\lambda)\varepsilon.\tag{F6}$$

Proof. When $\lambda>1$, the claim can be proved following exactly the same routine as Theorem 1. When $\lambda<1$, we follow the same routine as Theorem 1 to decompose the optimization problem into a two step minimization problem: fixed $\boldsymbol{w}_\mathrm{c}$, let $g(\mathbf{q},\dot{\mathbf{q}},t)=f(\mathbf{q},\dot{\mathbf{q}},t)-f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})$. Then, there exists a $\boldsymbol{w}_\mathrm{n}^g$, such that

$$\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_n^g)-g(\mathbf{q},\dot{\mathbf{q}},t)\|\le\varepsilon.$$

Therefore,

$$\begin{aligned}&\min_{\boldsymbol{w}_\mathrm{n}}\left(\|f(\mathbf{q},\dot{\mathbf{q}},t)-f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})-f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|+\lambda\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_\mathrm{n})\|\right)\\ \le&\|f(\mathbf{q},\dot{\mathbf{q}},t)-f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})-f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_n^g)\|+\lambda\|f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_n^g)\|\\ \le&(1+\lambda)\|f(\mathbf{q},\dot{\mathbf{q}},t)-f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c})-f_\mathrm{n}^{NN}(\mathbf{q},\dot{\mathbf{q}},t;\boldsymbol{w}_n^g)\|+\lambda\|f(\mathbf{q},\dot{\mathbf{q}},t)-f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c}))\|\\ \le&\lambda\|f(\mathbf{q},\dot{\mathbf{q}},t)-f_\mathrm{c}^{NN}(\mathbf{q},\dot{\mathbf{q}};\boldsymbol{w}_\mathrm{c}))\|+(1+\lambda)\varepsilon.\end{aligned}$$

Let $\boldsymbol{w}_{\mathrm{n}} = \arg\min_{\boldsymbol{w}_{\mathrm{n}}} \left(\|f(\mathbf{q}, \dot{\mathbf{q}}, t) - f_{\mathrm{c}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}; \boldsymbol{w}_{\mathrm{c}}) - f_{\mathrm{n}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}, t; \boldsymbol{w}_{\mathrm{n}})\| + \lambda \|f_{\mathrm{n}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}, t; \boldsymbol{w}_{\mathrm{n}})\| \right)$. Then,

$$
\begin{aligned}
&(1-\lambda)\|f(\mathbf{q}, \dot{\mathbf{q}}, t) - f_{\mathrm{c}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}; \boldsymbol{w}_{\mathrm{c}}) - f_{\mathrm{n}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}, t; \boldsymbol{w}_{\mathrm{n}})\| \\
\leq &\|f(\mathbf{q}, \dot{\mathbf{q}}, t) - f_{\mathrm{c}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}; \boldsymbol{w}_{\mathrm{c}}) - f_{\mathrm{n}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}, t; \boldsymbol{w}_{\mathrm{n}})\| + \lambda \|f_{\mathrm{n}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}, t; \boldsymbol{w}_{\mathrm{n}})\| \\
&-\lambda \|f(\mathbf{q}, \dot{\mathbf{q}}, t) - f_{\mathrm{c}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}; \boldsymbol{w}_{\mathrm{c}}))\| \\
\leq &(1+\lambda)\varepsilon,
\end{aligned} \tag{F7}
$$

which finishes the proof of Eq. (F5).
Let $h(\boldsymbol{w}_{\mathrm{c}}) = \min_{\boldsymbol{w}_{\mathrm{n}}} \|f(\mathbf{q}, \dot{\mathbf{q}}, t) - f_{\mathrm{c}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}; \boldsymbol{w}_{\mathrm{c}}) - f_{\mathrm{n}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}, t; \boldsymbol{w}_{\mathrm{n}})\| + \lambda \|f_{\mathrm{n}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}, t; \boldsymbol{w}_{\mathrm{n}})\|$. By Eq. (F7),

$$
\|h(\boldsymbol{w}_{\mathrm{c}})\| - \lambda \|f(\mathbf{q}, \dot{\mathbf{q}}, t) - f_{\mathrm{c}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}; \boldsymbol{w}_{\mathrm{c}})\| \leq (1+\lambda)\varepsilon,
$$

which further leads to

$$
\min_{\boldsymbol{w}_{\mathrm{c}}} \|h(\boldsymbol{w}_{\mathrm{c}})\| \leq (1+\lambda)\varepsilon + \lambda \min_{\boldsymbol{w}_{\mathrm{c}}} \|f(\mathbf{q}, \dot{\mathbf{q}}, t) - f_{\mathrm{c}}^{NN}(\mathbf{q}, \dot{\mathbf{q}}; \boldsymbol{w}_{\mathrm{c}})\|.
$$

This completes the proof. □