Integrating GNN and Neural ODEs for Estimating Non-Reciprocal Two-Body Interactions in Mixed-Species Collective Motion

Masahito Uwamichi[1*]**, Simon K. Schnyder**[2]**, Tetsuya J. Kobayashi**[2,3]**, Satoshi Sawai**[1,3]

[1]Graduate School of Arts and Sciences, The University of Tokyo
[2]Institute of Industrial Science, The University of Tokyo
[3]Universal Biology Institute, The University of Tokyo
`uwamichi@g.ecc.u-tokyo.ac.jp, simon@sat.t.u-tokyo.ac.jp,`
`tetsuya@sat.t.u-tokyo.ac.jp, cssawai@mail.ecc.u-tokyo.ac.jp`

Abstract

Analyzing the motion of multiple biological agents, be it cells or individual animals, is pivotal for the understanding of complex collective behaviors. With the advent of advanced microscopy, detailed images of complex tissue formations involving multiple cell types have become more accessible in recent years. However, deciphering the underlying rules that govern cell movements is far from trivial. Here, we present a novel deep learning framework for estimating the underlying equations of motion from observed trajectories, a pivotal step in decoding such complex dynamics. Our framework integrates graph neural networks with neural differential equations, enabling effective prediction of two-body interactions based on the states of the interacting entities. We demonstrate the efficacy of our approach through two numerical experiments. First, we used simulated data from a toy model to tune the hyperparameters. Based on the obtained hyperparameters, we then applied this approach to a more complex model with non-reciprocal forces that mimic the collective dynamics of the cells of slime molds. Our results show that the proposed method can accurately estimate the functional forms of two-body interactions – even when they are nonreciprocal – thereby precisely replicating both individual and collective behaviors within these systems.

1 Introduction

Collective motion, a phenomenon observed in various biological systems, is characterized by the coordinated movement of multiple entities. This behavior is prevalent in a wide range of self-propelled systems, collectively referred to as active matter (1), from flocks of birds (2; 3; 4) and schools of fish (5) to cellular slime molds (6), microswimmers (7), swarming leukocytes (8), and human crowds (9). Understanding the underlying mechanisms of collective motion is crucial for elucidating the principles governing the dynamics of these systems. In particular, the interactions between individual entities play a pivotal role in shaping the collective behavior of the system.

Recent advances in imaging technologies have enabled detailed observations at the cellular level, providing insights into the dynamics of complex tissue formation involving multiple cell types. For example, cellular slime molds, a model organism for studying collective motion, exhibit intricate behaviors such as aggregation, migration, and differentiation (6). These behaviors are driven by the interactions between different cell types, which are mediated by chemical signals and physical forces. Decoding the underlying equations of motion that govern these interactions is essential for understanding the emergent properties of these systems.

In this work, we present a novel deep learning framework for estimating two-body interactions in a mixed-species collective motion. Our framework integrates graph neural networks (GNNs) with neural differential equations (neural DEs) to predict interactions between pairs of entities based on their states. GNNs are well-suited for modeling complex interactions in graph-structured data, while neural DEs provide a flexible framework for learning the dynamics of the system. By combining these two approaches, we can effectively capture the interactions between individual entities and predict their collective behavior. We demonstrate the efficacy of our framework through two numerical experiments. The first experiment uses a toy model designed to generate data for refining the hyperparameters of our framework. The second experiment explores a complex scenario that partially mimics the collective motion of cellular slime molds, where two different cell types interact with each other. Our results show that our method can accurately estimate the two-body interactions, thereby replicating both individual and collective behaviors within these systems.

The rest of this paper is organized as follows. In section 2, we introduce the study of collective motion. In section 3, we provide an overview of related work on collective motion and deep learning for dynamical systems. In section 4, we describe our deep learning framework for estimating two-body interactions in mixed-species collective motion. In section 5, we successfully apply this framework to two numerical experiments. Finally, in section 6, we discuss the implications of our work and outline potential future research directions.

2 Background

Let us first describe the formulation of collective motion in active matter. Starting with the Vicsek model (10), the collective behavior in active matter is described based on the centroids, velocities, and orientations of each individual. A unique aspect of active matter is that it allows for spontaneous generation of forces and torques, which is justified by the ability to extract and utilize energy from the external environment (11). The Vicsek model itself is a multi-particle model in discrete time, where each individual possesses a velocity along its orientation and adjusts its direction based on interactions with nearby individuals. This model assumes that interactions among individuals are local and can be represented using a dynamical graph structure (12). Stemming from this model, numerous other models have been proposed, differing in the nature of interactions between individuals and the forms of their equations of motion. Particularly models with continuous-time motion equations often use Langevin equations where the motion of individuals is described by the summation of pairwise interactions (13). Furthermore, since active matter can utilize energy from the external environment, focusing solely on moving agents categorizes it as an open system. This implies that the conservation of the total system energy is not guaranteed, making these systems inherently unsuitable for Hamiltonian descriptions. Given this background, the formulation of collective motion in active matter typically involves direct descriptions of forces rather than using free energy.

3 Related Work

System Identification methods: This section discusses methods for estimating governing laws from data, commonly referred to as system identification. System identification aims to estimate the equations of motion of a system from data. One well-known method for estimating differential equations is Sparse Identification of Nonlinear Dynamics (SINDy) (14). SINDy estimates nonlinear differential equations from data using LASSO regression to sparsely estimate the terms of the equations. However, when applying this method to many-body systems with pairwise interactions, the number of parameters to estimate increases exponentially with the number of individuals, which raises computational costs and leads to potential instabilities. Other system identification methods use Bayesian optimization (15; 16; 17), e.g. representing the functions in the motion equations through basis function expansion, and estimating their coefficients via Bayesian optimization. This method has been successfully used in Vicsek model, reproducing its formation of orientational order. However, the effectiveness of this approach for more complex systems, such as mixed-species systems, remains unclear. Deep learning approaches to analyzing collective motion have also been proposed (18; 19; 20). These incorporate Attention mechanisms to analyze behaviors limited to turning right or left (18), systems that limit the scope to Vicsek-type models to estimate orientational order parameters (19), or to estimate two-body interactions by expressing forces as neural networks (20). Another possibility is to express Hamiltonians (21) or Lagrangians (22) as

neural networks, in order to preserve known symmetries of the studied system. Some proposed including knowledge or assumptions about the physical context into the loss function (23). So far, none of these methods have been applied to systems in which multiple species interact with each other in unknown ways.

Integration of graph neural network with neural differential equations: In this paper, we integrate GNN with neural ordinary differential equations (neural ODEs) to estimate interactions through evaluating equations of motion with dynamic edge structure. Most of the previous studies on integration of the two methods solve and train ODEs on a graph with fixed edges and consider changes in the edge weight (24; 25; 26; 27; 28). This approach requires a fully or almost fully connected graph for collective motion where the adjacency changes drastically over time, and causes memory inefficiency. On the other hand, some introduced a given time series of edge structure (29), making the system difficult to be extrapolated. Thus we resolved these issues by setting a rule to define edges at each time point instead of explicitly giving graph structure time series.

4 Method

We aim to estimate the rules of motion for individual entities within collective motion data. This section presents the general framework that is used to generate the training data from a physical model for two numerical experiments, as well as the learning algorithm and neural network architecture.

4.1 Physical model

We represent the state of each entity i at time t as $\mathcal{Z} \ni z^i(t) = (x^i(t), y^i) \in \mathcal{X} \times \mathcal{Y}$, where $x^i \in \mathcal{X}$ denotes the variables described by the motion equations in the state space $\mathcal{X} \subset \mathbb{R}^n$, and y^i, a non-temporal variable, represents auxiliary attributes such as the type of each entity within the feature space $\mathcal{Y}$. We define a distance function $d : \mathcal{Z} \times \mathcal{Z} \to \mathbb{R}$, and assume that entities i and j interact at time t if $d_{ij}(t) := d(z^i(t), z^j(t)) < d_0$, where $d_0 \in \mathbb{R}$ is a predefined threshold. The motion of each entity i is governed by the following Langevin equation:

$$dx^i = \left(F^{(1)}(z^i(t)) + \sum_{j \text{ s.t. } d_{ij}(t) < d_0 \wedge j \neq i} F^{(2)}(z^i(t), z^j(t)) \right) dt + \sigma dW^i(t), \tag{1}$$

with given self-driven forces $F^{(1)} : \mathcal{Z} \to \mathbb{R}^n$, forces due to interactions between pairs $F^{(2)} : \mathcal{Z} \times \mathcal{Z} \to \mathbb{R}^n$, intensity of noise $\sigma \in \mathbb{R}$, and a Wiener process $W^i(t) \in \mathbb{R}^n$. Given the initial state $Z(t_0) \in \mathcal{Z}$ at time $t_0 \in \mathbb{R}$, the state $Z(t_0 + \Delta t) \in \mathcal{Z}$ after a time interval $\Delta t \in \mathbb{R}$ can be determined by solving the motion equation using the operator $S(F^{(1)}, F^{(2)}, \sigma, d, d_0) : (Z(t_0), \Delta t) \mapsto Z(t_0 + \Delta t)$. Here, $z^i(t_0 + \Delta t) = (x^i(t_0 + \Delta t), y^i)$ is defined by the following integral:

$$x^i(t_0 + \Delta t) = x^i(t_0) + \int_{t_0}^{t_0 + \Delta t} dx^i(t), \tag{2}$$

which is computed numerically (section 4.3). For $\sigma = 0$, we evaluated eq.2 with the neural ODE method (30; 31), and for $\sigma \neq 0$ the neural stochastic differential equation (neural SDE) method (32; 33).

4.2 Interpretation of the system as a graph

The system at any time t can be represented as a directed graph $G(t) = (V, E(t), Z(t))$, where the set of entities represent the vertices $V = \{1, 2, \ldots, N\}$, pairs of interacting entities constitute edges $E(t) = \{(i, j) \in V \times V | d_{ij}(t) < d_0 \wedge i \neq j\}$, and $Z(t) = \{z^i(t) | i \in V\}$ is the set of states of all entities. Importantly, since the entities change states over time, the graph in general evolves over time. The forces $F^{(1)}$ and $F^{(2)}$ are evaluated on each vertex and each edge, respectively.

4.3 Solving the equations of motion of the graph through neural DEs

To obtain a time series of collective entities (eq. 2), we have to evaluate the equation of motion (eq. 1). We achieved this with a custom-made wrapper that connects neural ODE (SDE) to GNN (Figure
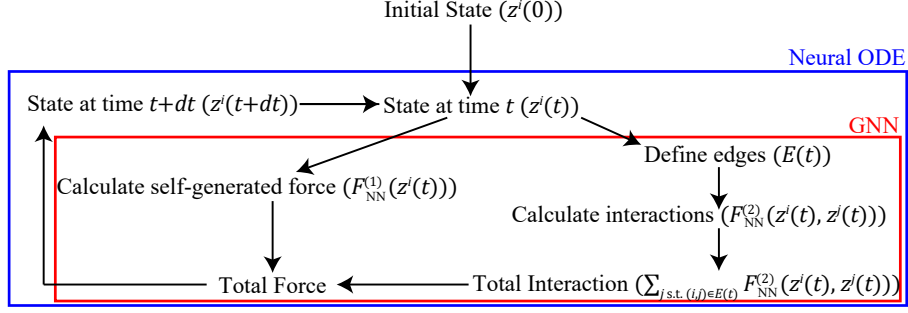
Figure 1: Schematic figure of the integration of GNN and neuralODE. For details, see section 4.3.

1). The wrapper is equipped with the distance function d and the threshold d_0 to define the edges $E(t)$ at each time t, the functions to calculate self-propulsion $F^{(1)}$ and interaction $F^{(2)}$, and the noise intensity σ. The computation $S(F^{(1)}, F^{(2)}, \sigma, d, d_0)$ is then run through the following steps:

0. Create a graph object only with nodes V and set the static states $\{y^i\}$ of the entities. Store this graph in the wrapper object. Pass the initial state $\{x^i(t_0)\}$ into the neural ODE (SDE) object (Figure 1 blue).

1. The neural ODE (SDE) object passes the current state $\{x^i(t)\}$ to the *forward* function of the wrapper object.

2. The *forward* function places the state values $\{x^i(t)\}$ at the nodes of the graph. Subsequently, the graph is passed to GNN (Figure 1 red).

3. GNN first defines and updates the edges $E(t)$ using the distance function d and the threshold d_0 stored in the wrapper. Next, it calculates the interactions $F^{(2)}$ on the edges and adds them to the self-propulsion $F^{(1)}$ on the nodes to return the forces to the ODE (SDE) object. If SDE, GNN also returns the noise intensity σ in the wrapper.

4. The neural ODE (SDE) object updates the state $\{x^i(t + dt)\}$.

5. Repeat steps 1-4 until the end of the time series.

When inferring the forces $F^{(1)}, F^{(2)}$, we model them using neural networks (section 4.5).

4.4 Generation of training data

To construct $S_{\text{sim}} = S(F_{\text{sim}}^{(1)}, F_{\text{sim}}^{(2)}, \sigma_{\text{sim}}, d_{\text{sim}}, d_{0,\text{sim}})$, distance functions d_{sim} were provided, suitable choices for the noise magnitude σ_{sim}, threshold $d_{0,\text{sim}}$ and forms for the forces $F_{\text{sim}}^{(1)}, F_{\text{sim}}^{(2)}$ were specified (see section 5). Thus defined S_{sim} was used to conduct simulations according to the motion equations, from initial states assigned uniformly at random $x^i(0) \in \mathcal{X}_0$, with appropriate values for $y^i \in \mathcal{Y}$. The simulation was repeated M times, and the m-th result is denoted as $Z_{\text{sim}}^m : t \mapsto S_{\text{sim}}(Z_{\text{sim}}^m(0), t)$ with the m-th initial condition $Z_{\text{sim}}^m(0)$. The combined data set $Z_{\text{sim}} = \{Z_{\text{sim}}^m | m = 1, \ldots, M\}$ was then used to apply the subsequent learning algorithm, aiming to estimate $F_{\text{sim}}^{(1)}$ and $F_{\text{sim}}^{(2)}$.

4.5 Learning algorithm and neural network architecture for the forces

Subsequently, we developed a learning algorithm to estimate $F_{\text{sim}}^{(1)}$ and $F_{\text{sim}}^{(2)}$ using the results from the simulations conducted.

The self-propulsion and interaction functions, $F_{\text{NN}}^{(1)}$ and $F_{\text{NN}}^{(2)}$, were modeled using neural networks. Unless otherwise specified, these functions consisted of a three-layer fully connected network followed by a scaling layer. Each fully connected layer consisted of 128 nodes, with the Exponential Linear Unit (ELU) serving as the activation function. The scaling layer involved a scalar $A \in \mathbb{R}$ and a vector $B \in \mathbb{R}^n$ as learnable parameters, and it transformed an input vector $\alpha \in \mathbb{R}^n$ to $e^A \alpha + B$. This
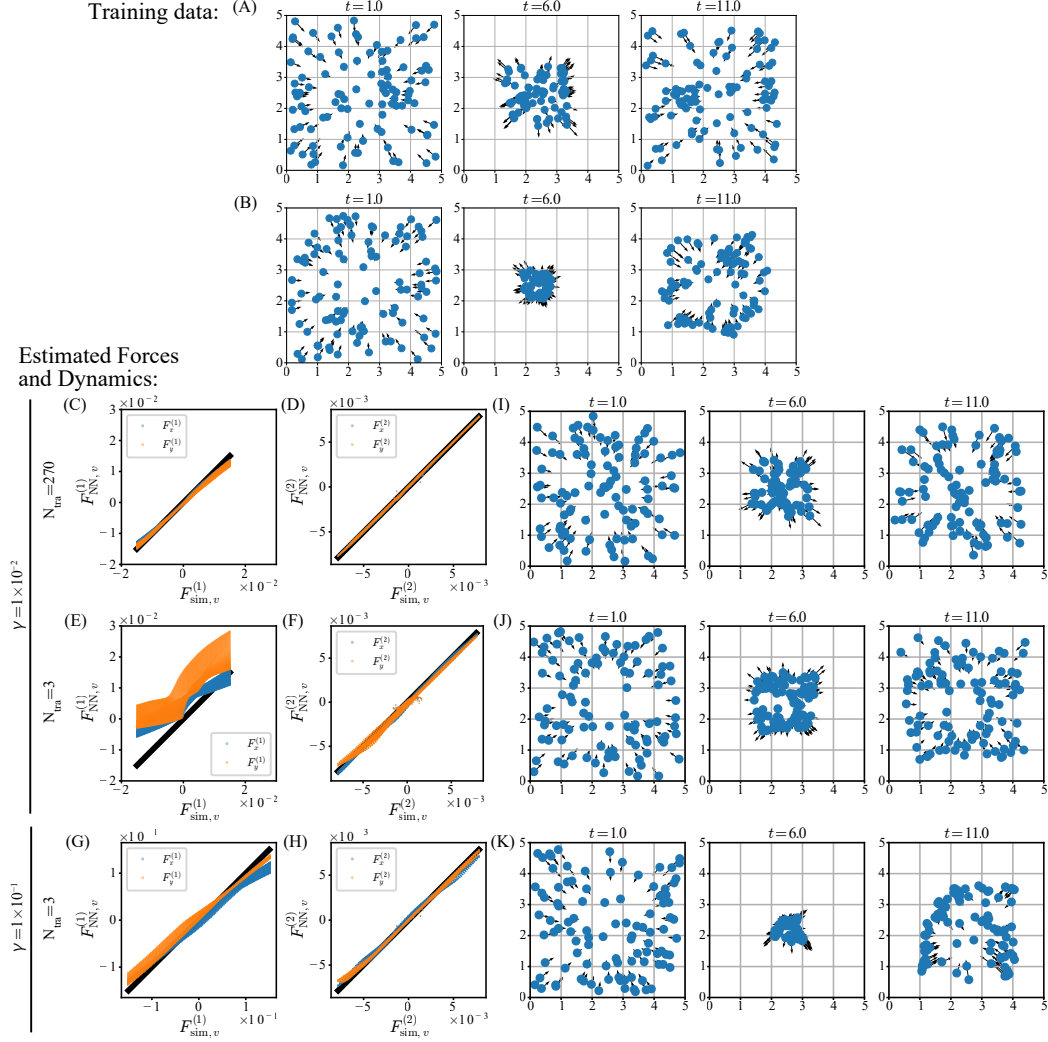
Figure 2: Snapshots of the simulation results in the harmonic interaction model with friction constants. (A) The case with a friction constant of $\gamma = 1 \times 10^{-2}$ and (B) with $\gamma = 1 \times 10^{-1}$. Positions of individuals are indicated by blue circles, and velocities by black arrows. (C-H) The functions estimated from the data with harmonic interaction plotted against the true values. The upper panel displays $F^{(1)}$ and the lower panel $F^{(2)}$ for different cases: (C-D) with $\gamma = 1 \times 10^{-2}$ and $N_{\text{tra}} = 270$, (E-F) with $\gamma = 1 \times 10^{-2}$ and $N_{\text{tra}} = 3$, and (G-H) with $\gamma = 1 \times 10^{-1}$ and $N_{\text{tra}} = 3$. Blue and orange indicates x- or y- element F. A black line serves as a guide indicating where the estimated values equal the true values. (I-K) Snapshots of the simulation results in the model estimated from the data with harmonic interaction. The panels represent different scenarios: (I) with $\gamma = 1 \times 10^{-2}$ and $N_{\text{tra}} = 270$, (J) with $\gamma = 1 \times 10^{-2}$ and $N_{\text{tra}} = 3$, and (K) with $\gamma = 1 \times 10^{-1}$ and $N_{\text{tra}} = 3$.

configuration allowed the output of the fully connected network to be scaled appropriately, reflecting physical scales.

The parameters of fully connected networks and the scaling layer, collectively denoted θ, were optimized to approximate $F_{\text{NN}}^{(1)}(\theta)$ and $F_{\text{NN}}^{(2)}(\theta)$ to $F_{\text{sim}}^{(1)}$ and $F_{\text{sim}}^{(2)}$, respectively. To evaluate the deviation of $F_{\text{NN}}^{(1)}(\theta)$ and $F_{\text{NN}}^{(2)}(\theta)$, we first solved the motion equations using these functions. In practice, we constructed a wrapper object $S_{\text{NN}}(\theta) = S(F_{\text{NN}}^{(1)}(\theta), F_{\text{NN}}^{(2)}(\theta), 0, d_{\text{sim}}, d_{0,\text{NN}})$ and used it to perform simulations. Here, $d_{0,\text{NN}}$ was a suitably set threshold, and for simplicity, $\sigma_{\text{NN}} = 0$. In the simulation, the initial state was set as a snapshot in training data $Z_{\text{sim}}^{m}(t_0)$, and we used this

5

to compute the state τ time units later as $Z_{NN}^m(\theta; t_0 + \tau) = S_{NN}(\theta)(Z_{sim}^m(t_0), \tau)$, which would be compared with a truth $Z_{sim}^m(t_0 + \tau)$.

For each numerical experiment, a loss function $\mathcal{L} : \mathcal{Z} \times \mathcal{Z} \to \mathbb{R}$ was used to evaluate the discrepancy between the solution by θ and the simulation data through $\mathcal{L}^m(\theta; t_0) = \mathcal{L}(Z_{sim}^m(t_0 + \tau), Z_{NN}^m(\theta; t_0 + \tau))$. To optimize θ, the average $\mathcal{L}(\theta)$ of this loss function over various m and t_0 was minimized:

$$\theta^* = \arg\min_\theta \mathcal{L}(\theta) \tag{3}$$

The M simulation data sets were split into two parts, with M_{tra} sets used as training data and M_{val} sets used as validation data. This optimization was conducted using the LAMB optimization algorithm (34), leveraging gradient information with respect to θ. The parameter θ^* was evaluated using the minimum of the loss function calculated on the validation data set.

5 Experiments

Here, we demonstrate the efficacy of the proposed method in two numerical experiments.

5.1 Underdamped Brownian Motion with Harmonic Interaction

Training data: First, to demonstrate that the self-propulsion and interaction forces could be estimated using our method, we performed simulations using a toy model. We used $N = 100$ particles, each with position $r^i(t) \in \mathbb{R}^2$ and velocity $v^i(t) \in \mathbb{R}^2$, such that $z^i(t) = x^i(t) = (r^i(t), v^i(t))$ and y^i is empty. The particles were connected by harmonic oscillators, and each velocity was subjected to a damping force, so the equations of motion read:

$$dr^i = v^i dt, \tag{4}$$

$$dv^i = \left(-\gamma v^i - \sum_{j\,\text{s.t.}\,(i,j)\in E(t)} \nabla_{r^i} U(r^i - r^j) \right) dt + \sigma dW^i(t), \tag{5}$$

$$U(r) = \frac{1}{2}k(|r| - r_c)^2, \tag{6}$$

where $k \in \mathbb{R}$ is the strength of the interaction, $r_c \in \mathbb{R}$ the natural spring length, and $\gamma \in \mathbb{R}$ the friction coefficient. The distance function was $d_{sim}(z^i, z^j) = |r^i - r^j|$, with a threshold set at $d_{0,sim} = 5$.

In order to fit these terms into the aforementioned framework, the equations were reformulated as:

$$F_{sim}^{(1)}(z^i) = (F_{sim,r}^{(1)}(z^i), F_{sim,v}^{(1)}(z^i)) = (v^i, -\gamma v^i), \tag{7}$$

$$F_{sim}^{(2)}(z^i, z^j) = (F_{sim,r}^{(2)}(z^i, z^j), F_{sim,v}^{(2)}(z^i, z^j)) = (0, -\nabla_{r^i} U(r^i - r^j)). \tag{8}$$

Here, $k = 1 \times 10^{-3}$, $r_c = 1$, $\sigma = 1 \times 10^{-3}$, and γ was set to 1×10^{-2} or 1×10^{-1}. The initial states $x^i(0)$ were uniformly sampled from $\mathcal{X}_0 = [0, 5]^2 \times [0, 1 \times 10^{-3}]^2$, and simulations were performed to obtain Z_{sim}.

For these simulations, the Euler-Maruyama method with a time step of 0.1 was used, and data were collected at $t = 0, 1, \ldots, 50$. The data showed that all particles cyclically gathered towards the center before dispersing (Figure 2(A-B)). When $\gamma = 1 \times 10^{-2}$, the amplitude of aggregation-dispersion was nearly constant (Figure 2(A), Supplemental Movie S1), whereas for $\gamma = 1 \times 10^{-1}$, the amplitude of aggregation-dispersion decreased (Figure 2(B), Supplemental Movie S2).

Learning: With these data, we applied our proposed method to estimate $F_{sim}^{(1)}$ and $F_{sim}^{(2)}$. $F_{NN}^{(1)}$ and $F_{NN}^{(2)}$ were modeled as follows:

$$F_{NN}^{(1)}(z^i; \theta) = (v^i, F_{NN,v}^{(1)}(v^i; \theta)), \tag{9}$$

$$F_{NN}^{(2)}(z^i, z^j; \theta) = (0, F_{NN,v}^{(2)}(r^j - r^i; \theta)). \tag{10}$$

$F_{NN,v}^{(1)}$ and $F_{NN,v}^{(2)}$ were each independent neural networks as described in section 4.5. We evaluated the predictive accuracy for position and velocity as follows, using $Z_{NN}^m(\theta; t_0 + \tau)$ predicted for $\tau = 10$

and the simulation data $Z_{\mathrm{sim}}^m(t_0)$:

$$\mathcal{L}_r(Z_{\mathrm{sim}}^m(t_0 + \tau), Z_{\mathrm{NN}}^m(\theta; t_0 + \tau)) \quad = \quad \frac{1}{N} \sum_{i \in V} |r_{\mathrm{NN}}^i(\theta; t_0 + \tau) - r_{\mathrm{sim}}^i(t_0 + \tau)|^2, \qquad (11)$$

$$\mathcal{L}_v(Z_{\mathrm{sim}}^m(t_0 + \tau), Z_{\mathrm{NN}}^m(\theta; t_0 + \tau)) \quad = \quad \frac{1}{N} \sum_{i \in V} |v_{\mathrm{NN}}^i(\theta; t_0 + \tau) - v_{\mathrm{sim}}^i(t_0 + \tau)|^2. \qquad (12)$$

We sampled these metrics for 60 randomly selected pairs (m, t_0). The total loss function $\mathcal{L}(\theta)$ for one batch was defined as the dimensionless sum of these metrics, normalized by the variance in the simulation data:

$$\mathcal{L}_r(\theta) \quad = \quad \sum_{(m,t_0)} \mathcal{L}_r(Z_{\mathrm{sim}}^m(t_0 + \tau), Z_{\mathrm{NN}}^m(\theta; t_0 + \tau))/\mathrm{Var}_{i,(m,t_0)}[r_{\mathrm{sim}}^i(t_0 + \tau)], \qquad (13)$$

$$\mathcal{L}_v(\theta) \quad = \quad \sum_{(m,t_0)} \mathcal{L}_v(Z_{\mathrm{sim}}^m(t_0 + \tau), Z_{\mathrm{NN}}^m(\theta; t_0 + \tau))/\mathrm{Var}_{i,(m,t_0)}[v_{\mathrm{sim}}^i(t_0 + \tau)], \qquad (14)$$

$$\mathcal{L}(\theta) \quad = \quad \mathcal{L}_r(\theta) + \mathcal{L}_v(\theta). \qquad (15)$$

To minimize the loss function, we manually searched for optimal hyperparameters for the LAMB optimizer. We determined that $\beta_1 = 0.5$, $\beta_2 = 0.4$, $\epsilon = 1 \times 10^{-6}$, and a learning rate of 1×10^{-3} effectively minimized the loss function, although convergence was notably slow. It is important to note that we were unable to identify hyperparameters that would allow for both faster convergence and adequate estimation of the functional forms of $F^{(1)}$ and $F^{(2)}$. As a result of minimizing this loss function, we optimized θ to estimate $F_{\mathrm{NN}}^{(1)}$ and $F_{\mathrm{NN}}^{(2)}$ (Figure 2 (C-H)). When $\gamma = 1 \times 10^{-2}$ and the training set size was $M_{\mathrm{tra}} = 270$ and validation set size $M_{\mathrm{val}} = 30$, after 300 epochs (15 days), $F_{\mathrm{NN},v}^{(1)}$ and $F_{\mathrm{NN},v}^{(2)}$ were observed to approximate $F_{\mathrm{sim},v}^{(1)}$ and $F_{\mathrm{sim},v}^{(2)}$ respectively (Figure 2(C-D)). In contrast, with $M_{\mathrm{tra}} = 3$ and $M_{\mathrm{val}} = 3$, especially $F_{\mathrm{NN},v}^{(1)}$ did not approximate $F_{\mathrm{sim},v}^{(1)}$ even after 6000 epochs (4 days; Figure 2(E-F)). However, under the same dataset size but with $\gamma = 1 \times 10^{-1}$, $F_{\mathrm{NN},v}^{(1)}$ and $F_{\mathrm{NN},v}^{(2)}$ were confirmed to approximate $F_{\mathrm{sim},v}^{(1)}$ and $F_{\mathrm{sim},v}^{(2)}$ (Figure 2(G-H)).

The accuracy of the estimation results is quantified in Supplemental Table S1. For each particle or pair in the dataset Z_{sim}, we calculated the forces $F_v^{(1)}$ and $F_v^{(2)}$ and computed the Mean Squared Error (MSE) and Mean Absolute Error (MAE). These errors were then normalized by the L1 or L2 norm of $F_{\mathrm{sim},v}^{(1)}$ and $F_{\mathrm{sim},v}^{(2)}$ respectively, to provide dimensionless measures of accuracy. Due to the extensive computation time required for estimation, comprehensive statistics could not be gathered. Instead, we present the results of all trials to illustrate the trends in estimation accuracy. As observed, when $M_{\mathrm{tra}} = 3$ with a low friction constant $\gamma = 1 \times 10^{-2}$, significant estimation errors occurred. Conversely, the estimation errors for the interaction force $F_v^{(2)}$ were largely unaffected by γ, suggesting that the estimation of interaction and friction are somewhat independent. However, when $M_{\mathrm{tra}} = 270$, a slight improvement in the accuracy of interaction estimates was noted compared to the $M_{\mathrm{tra}} = 3$ scenario. This observation indicates that the accuracy of estimation depends on the number of data points, for the interaction as well as the friction.

Furthermore, to verify how well these estimates fit the training data, we conducted simulations from random initial values $Z(0)$ sampled in a similar manner to training data creation, and visualized the results $S_{\mathrm{NN}}(\theta^*)(Z(0), t)$ (Figure 2(I-K); Supplemental Movies S3-5). In all cases, the estimates were confirmed to adequately replicate the training data for all cases, even with $\gamma = 1 \times 10^{-2}$ and $M_{\mathrm{tra}} = 3$. The ability to replicate training data suggests that friction had almost no effect in this case. However, this implies that the proposed method may not adequately estimate very weak effects.

5.2 Mixed Species Collective Motion with Overdamped Self-propulsion

Next, to test the proposed method in complex systems involving interactions among multiple species, we conducted simulations using a more complex model that emulates real collective movements.

Training data: In this model, each of the $N = 400$ individuals has position $r^i(t) \in [0, L]^2$, orientation $\phi^i(t) \in [0, 2\pi]$, and species type $c^i \in \{0, 1\}$. Both r^i and ϕ^i are subject to periodic
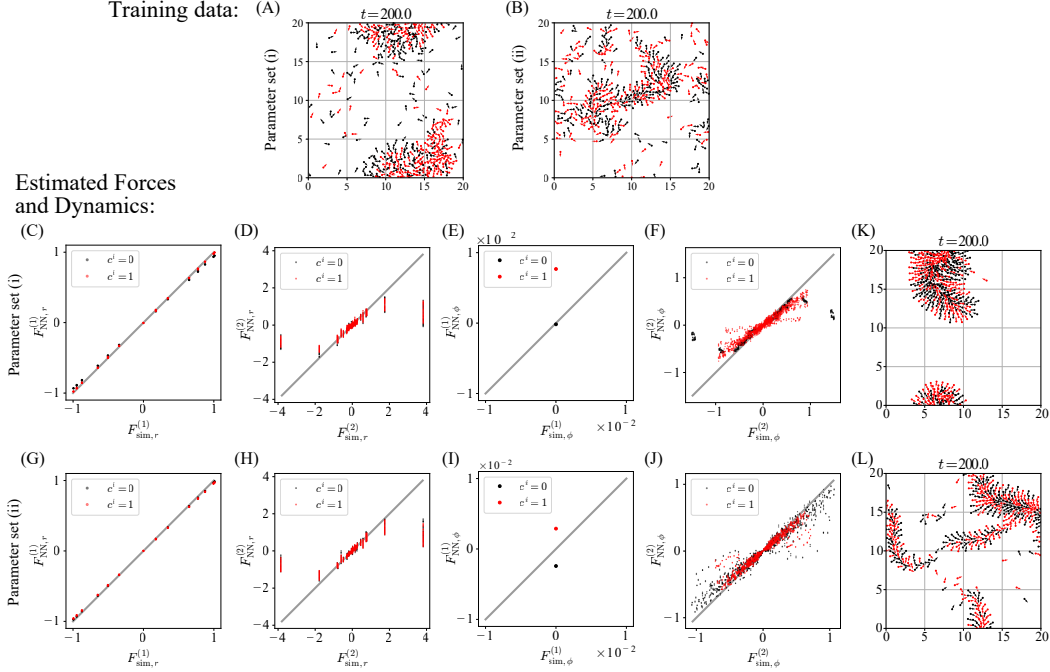
Figure 3: Snapshots of the simulation results in the mixed-species model. Panel (A) displays a representative snapshot of the training data for the case with parameter set (i), and panel (B) with parameter set (ii) (see Supplemental Table S2). Each individual is represented by an arrow located at their position and directed toward their polarity. Black arrows indicate individuals with $c^i = 0$, while red arrows represent those with $c^i = 1$. (C-J) The functions estimated from the data with the mixed-species model plotted against the true values. The rows display different cases: (C-F) with parameter set (i), and (G-J) with parameter set (ii). A gray line serves as a guide indicating where the estimated values equal the true values. (K-L) Snapshots of the simulation results in the model estimated from data with the mixed-species model. (K) The case with parameter set (i), and (L) with parameter set (ii).

boundary conditions, ensuring continuous and consistent movement dynamics across the defined space. This defines $x^i(t) = (r^i(t), \phi^i(t))$ and $y^i = (c^i)$, constituting $z^i(t) = (x^i(t), y^i)$. The motion equations for individual i are described as follows:

$$dr^i = (v_0 p^i + \sum_{j \, \mathrm{s.t.} \, (i,j) \in E(t)} \beta J_{\mathrm{eV}}^{ij}) dt, \tag{16}$$

$$d\phi^i = - \sum_{j \, \mathrm{s.t.} \, (i,j) \in E(t)} \left(\alpha_{\mathrm{CF}}(c^i) J_{\mathrm{CF}}^{ij} + \alpha_{\mathrm{Ch}}(c^i) J_{\mathrm{Ch}}^{ij} \right) \left(r^{ij} \cdot p_\perp^i \right) dt + \sigma dW^i(t), \tag{17}$$

$$J_{\mathrm{eV}}^{ij} = \left(r_c^{-1} - |r^{ij}|^{-1} \right) r^{ij}, \tag{18}$$

$$J_{\mathrm{CF}}^{ij} = \frac{1}{2}(1 - \frac{r^{ij} \cdot p^j}{|r^{ij}|}), \tag{19}$$

$$J_{\mathrm{Ch}}^{ij} = - \frac{r^{ij} \cdot p^i}{|r^{ij}|} K_1(\kappa |r^{ij}|), \tag{20}$$

$$p^i = (\cos \phi^i, \sin \phi^i), \ \ p_\perp^i = (-\sin \phi^i, \cos \phi^i), \tag{21}$$

$$r^{ij} = r^j - r^i \in [-L/2, L/2]^2, \tag{22}$$

with self-propulsion speed $v_0 \geq 0$, the strength of the excluded volume interaction $\beta \geq 0$, the strengths of contact following and chemotaxis $\alpha_{\mathrm{CF}}, \alpha_{\mathrm{Ch}} \geq 0$, respectively, the diffusion length of the chemoattractant $\kappa \geq 0$, the noise strength $\sigma \geq 0$, and the modified Bessel function of the second kind K_1. The model was obtained by introducing a chemotaxis term into a preexisting model (35), to make it more appropriate for cellular slime molds. The chemotaxis term assumes rapid diffusion

of chemotactic substances secreted by each individual (36). We varied the sensitivity of contact following and chemotaxis depending on the species type, thus modeling a system where different species interact non-reciprocally. The interaction terms were adapted into our framework as follows:

$$F_{\text{sim}}^{(1)}(z^i) = (F_{\text{sim},r}^{(1)}(z^i), F_{\text{sim},\phi}^{(1)}(z^i)) \tag{23}$$

$$= (v_0 p^i, 0), \tag{24}$$

$$F_{\text{sim}}^{(2)}(z^i, z^j) = (F_{\text{sim},r}^{(2)}(z^i, z^j), F_{\text{sim},\phi}^{(2)}(z^i, z^j)) \tag{25}$$

$$= (\beta J_{\text{eV}}^{ij}, -(\alpha_{\text{CF}}(c^i) J_{\text{CF}}^{ij} - \alpha_{\text{Ch}}(c^i) J_{\text{Ch}}^{ij}) (r^{ij} \cdot p_\perp^i)). \tag{26}$$

The distance function was $d_{\text{sim}}(z^i, z^j) = |r^{ij}|$, with periodic boundary conditions of width $L = 20$ and a threshold $d_{0,\text{sim}} = 1$. The parameters were $v_0 = 1$, $\beta = 1$, $\kappa = 0.5$, and $\sigma = \sqrt{0.2}$, with 200 individuals of each species $c^i = 0, 1$. Parameter dependencies based on c^i are shown in Supplemental Table S2.

The initial states $x^i(0)$ were uniformly sampled from $[0, L]^2 \times [0, 2\pi]$, and simulations were carried out to generate Z_{sim} using the Heun method with a timestep of 0.1, collecting data for $t = 0, 1, \ldots, 300$. Simulation results for parameter set (i) showed the formation of mixed-species circular clusters exhibiting rotational behavior (Figure 3(A), Supplemental Movie S6). Conversely, results for parameter set (ii) formed centipede-like clusters of mixed species that moved translationally in local alignments (Figure 3(B), Supplemental Movie S7).

Learning: In order to estimate $F_{\text{sim}}^{(1)}$ and $F_{\text{sim}}^{(2)}$, we modeled $F_{\text{NN}}^{(1)}$ and $F_{\text{NN}}^{(2)}$ as follows:

$$F_{\text{NN}}^{(1)}(z^i; \theta) = (\mathcal{R}(\phi^i) F_{\text{NN},r}^{(1)}(c^i; \theta), F_{\text{NN},\phi}^{(1)}(c^i; \theta)), \tag{27}$$

$$F_{\text{NN}}^{(2)}(z^i, z^j; \theta) = \begin{pmatrix} \mathcal{R}(\phi^i) F_{\text{NN},r}^{(2)}(\mathcal{R}(-\phi^i) r^{ij}, \phi^j - \phi^i, c^i, c^j; \theta) \\ F_{\text{NN},\phi}^{(2)}(\mathcal{R}(-\phi^i) r^{ij}, \phi^j - \phi^i, c^i, c^j; \theta) \end{pmatrix}^T. \tag{28}$$

Here, $F_{\text{NN},r}^{(1)}, F_{\text{NN},\phi}^{(1)}, F_{\text{NN},r}^{(2)}, F_{\text{NN},\phi}^{(2)}$ are independent neural networks with their respective parameters, and $\mathcal{R}(\phi^i)$ is the rotation matrix for ϕ^i, which converts all angles into relative angles with respect to ϕ^i, thereby maintaining the model's invariance to coordinate rotation. Additionally, when inputting the categorical variable c^i into the neural network, it is transformed into a two-dimensional vector via an embedding layer. The predictions $Z_{\text{NN}}^m(\theta; t_0 + \tau)$ made by this neural network for $\tau = 1$ and the simulation data $Z_{\text{sim}}^m(t_0)$ defined the prediction errors for position and orientation as follows:

$$\mathcal{L}_r(Z_{\text{sim}}^m(t_0 + \tau), Z_{\text{NN}}^m(\theta; t_0 + \tau)) = \frac{1}{N} \sum_{i \in V} d(r_{\text{NN}}^i(\theta; t_0 + \tau), r_{\text{sim}}^i(t_0 + \tau))^2, \tag{29}$$

$$\mathcal{L}_\phi(Z_{\text{sim}}^m(t_0 + \tau), Z_{\text{NN}}^m(\theta; t_0 + \tau)) = \frac{1}{N} \sum_{i \in V} \left(1 - \cos(\phi_{\text{NN}}^i(\theta; t_0 + \tau) - \phi_{\text{sim}}^i(t_0 + \tau))\right). \tag{30}$$

Predictions were computed using the Euler-Maruyama method with a timestep of 0.1. These metrics were sampled for 60 randomly selected pairs (m, t_0) and each was normalized by the variance in the simulation data to compose the loss function for one batch:

$$\mathcal{L}_r(\theta) = \sum_{(m,t_0)} \mathcal{L}_r(Z_{\text{sim}}^m(t_0 + \tau), Z_{\text{NN}}^m(\theta; t_0)) / \text{Var}_{i,(m,t_0)}[r_{\text{sim}}^i(t_0 + \tau)], \tag{31}$$

$$\mathcal{L}_\phi(\theta) = \sum_{(m,t_0)} \mathcal{L}_\phi(Z_{\text{sim}}^m(t_0 + \tau), Z_{\text{NN}}^m(\theta; t_0)) / \sum_{i,(m,t_0)} \left(1 - \cos(\phi_{\text{sim}}^i(t_0 + \tau))\right), \tag{32}$$

$$\mathcal{L}(\theta) = \mathcal{L}_r(\theta) + \mathcal{L}_\phi(\theta). \tag{33}$$

To minimizing this loss function, we tried the same hyperparameters as those used for the harmonic interaction model, and found we optimized θ to estimate $F_{\text{NN}}^{(1)}$ and $F_{\text{NN}}^{(2)}$ (Figure 3(C-J)). All experiments were conducted with $M_{\text{tra}} = 3$, $M_{\text{val}} = 3$. After 20,000 epochs (25 days), $F_{\text{NN},r}^{(1)}, F_{\text{NN},\phi}^{(1)}, F_{\text{NN},r}^{(2)}, F_{\text{NN},\phi}^{(2)}$ were found to approximate $F_{\text{sim},r}^{(1)}, F_{\text{sim},\phi}^{(1)}, F_{\text{sim},r}^{(2)}, F_{\text{sim},\phi}^{(2)}$ closely (Figure 3(C-J)). To further verify the fit of our estimation results with the training data, we sampled random initial values $Z(0)$ similarly to the

training data creation process and conducted simulations to visualize the outcomes $S_{\mathrm{NN}}(\theta^*)(Z(0), t)$ (Figure 3(K-L); Supplemental Movies S8-9). The results confirmed that all estimates adequately reproduced the training data.

To further test the predictive power of the model trained on the mixed state, we performed simulations where all individuals were of the same type. For initial conditions $Z_c(0)$ where all $N = 400$ individuals shared the same type $c^i = c$, either $c = 0$ or $c = 1$, we performed simulations using the aforementioned model S_{sim} and the neural network $S_{\mathrm{NN}}(\theta^*)$ trained above (Supplemental Figure S1). For parameter set (i) with $c = 0$, both $S_{\mathrm{sim}}(Z_0(0), t)$ and $S_{\mathrm{NN}}(\theta^*)(Z_0(0), t)$ showed the formation of two clusters that exhibited rotational movements (Supplemental Figure S1(A-B)). Conversely, for parameter set (i) with $c = 1$ and parameter set (ii) with $c = 0, 1$, both $S_{\mathrm{sim}}(Z_0(0), t)$ and $S_{\mathrm{NN}}(\theta^*)(Z_0(0), t)$ formed centipede-like clusters that moved translationally in alignment with their local direction (Supplemental Figure S1(C-H)). These results suggest that the interaction of each type was approximated accurately enough to explain the dynamics of training data.

As a negative control, we conducted simulations where both types of individuals shared the same parameters, as in parameter sets (iii)-(v), and trained the neural networks for each case (Supplemental Figure S2). In all instances, it was confirmed that $F_{\mathrm{NN},r}^{(1)}, F_{\mathrm{NN},\phi}^{(1)}, F_{\mathrm{NN},r}^{(2)}, F_{\mathrm{NN},\phi}^{(2)}$ closely approximated $F_{\mathrm{sim},r}^{(1)}, F_{\mathrm{sim},\phi}^{(1)}, F_{\mathrm{sim},r}^{(2)}, F_{\mathrm{sim},\phi}^{(2)}$. Notably, these estimation results showed minimal dependency on c^i in $F_{\mathrm{NN}}^{(1)}(z^i; \theta^*), F_{\mathrm{NN}}^{(2)}(z^i, z^j; \theta^*)$. This outcome suggests that the species dependency of interactions estimated by the proposed method in mixed assemblies is based on actual data, rather than being an artifact of the methodology.

Finally, to summarize the results above, we quantified the estimation accuracy for each trial, similar to the methods described in section 5.1 (Supplemental Table S3). To note, since $F_{\mathrm{sim},\phi}^{(1)} = 0$, the MSE and MAE for $F_{\phi}^{(1)}$ are not normalized. Our results demonstrate that $F_r^{(1)}$ is estimated with high accuracy across all conditions. Also, the estimation of $F_{\phi}^{(1)}$ is maintained at sufficiently small magnitudes relative to the order of magnitude (10^0) typically seen with $F_{\phi}^{(2)}$ (Figure 3(F,J)). Regarding the interactions, both $F_r^{(2)}$ and $F_{\phi}^{(2)}$ exhibited somewhat higher errors, approximately of the order of 10^{-1}. These errors likely stem from the high dimensionality of the inputs to the $F_{\mathrm{NN}}^{(2)}$, which may prevent the neural network from achieving sufficient learning, or from the training process neglecting the noise.

6 Conclusion

In this study, we proposed a novel method for estimating interactions among individuals within models of collective motion. This method combines dynamic GNNs with neural ODEs to estimate interactions among individuals. We demonstrated that this method was able to estimate interactions in both simple models and complex mixed-species collective motion models. In models with simple harmonic interactions, the proposed method identified the relevant interactions effectively. Importantly, the method successfully inferred the non-reciprocal interactions between the different species in the complex mixed-species model for the collective dynamics of slime molds. For the latter case, the continuous updating of the edge structures in our approach substantially reduces memory needs. In our simulations of 400 bodies with an edge density of 2%, the required memory of 200 GB for a fully connected graph is reduced to 30GB thus making them feasible on off-the-shelf GPUs. Nevertheless, we should note that this approach is limited due to the long time required for estimation. Moreover, the method currently estimates deterministic motion equations and only considers pairwise interactions, thereby not accommodating interactions among three or more bodies or the effects of noise. Future research is expected to extend this approach to estimate more general interactions and develop methods for stochastic motion equations. Applying the present methods to real data in systems such as immune cells should help clarify the complex rules behind their migration stategies.

Acknowledgements

References

[1] T. Vicsek and A. Zafeiris, "Collective motion," *Physics Reports*, vol. 517, no. 3-4, pp. 71–140, 2012.

[2] A. Cavagna, A. Cimarelli, I. Giardina, G. Parisi, R. Santagati, F. Stefanini, and M. Viale, "Scale-free correlations in starling flocks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 107, pp. 11865–11870, 6 2010.

[3] A. Flack, M. Nagy, W. Fiedler, I. D. Couzin, and M. Wikelski, "From local collective behavior to global migratory patterns in white storks," *Science*, vol. 360, no. 6391, pp. 911–914, 2018.

[4] H. J. Charlesworth and M. S. Turner, "Intrinsically motivated collective motion," *Proceedings of the National Academy of Sciences*, vol. 116, pp. 15362–15367, 2019.

[5] Y. Katz, K. Tunstrøm, C. C. Ioannou, C. Huepe, I. D. Couzin, S. A. Levin, I. D. C. Designed, and I. D. C. Performed, "Inferring the structure and dynamics of interactions in schooling fish," *Proceedings of the National Academy of Sciences*, vol. 108, pp. 18720–18725, 2011.

[6] T. Fujimori, A. Nakajima, N. Shimada, and S. Sawai, "Tissue self-organization based on collective cell migration by contact activation of locomotion and chemotaxis," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 116, pp. 4291–4296, Mar. 2019.

[7] J. Elgeti, R. G. Winkler, and G. Gompper, "Physics of microswimmers - single particle motion and collective behavior: A review," *Reports on Progress in Physics*, vol. 78, 5 2015.

[8] E. Strickland, D. Pan, C. Godfrey, J. S. Kim, A. Hopke, W. Ji, M. Degrange, B. Villavicencio, M. K. Mansour, C. S. Zerbe, D. Irimia, A. Amir, and O. D. Weiner, "Self-extinguishing relay waves enable homeostatic control of human neutrophil swarming," *Developmental Cell*, July 2024.

[9] V. J. Kok, M. K. Lim, and C. S. Chan, "Crowd behavior analysis: A review where physics meets biology," *Neurocomputing*, vol. 177, pp. 342–362, 2016.

[10] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet, "Novel Type of Phase Transition in a System of Self-Driven Particles," *Physical Review Letters*, vol. 75, pp. 1226–1229, Aug. 1995.

[11] G. I. Menon, *Active Matter*, pp. 193–218. New York, NY: Springer New York, 2010.

[12] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D. Hwang, "Complex networks: Structure and dynamics," *Physics Reports*, vol. 424, pp. 175–308, Feb. 2006.

[13] H. Chaté, F. Ginelli, G. Grégoire, and F. Raynaud, "Collective motion of self-propelled particles interacting without cohesion," *Physical Review E*, vol. 77, no. 4, p. 046113, 2008.

[14] 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 of the United States of America*, vol. 116, pp. 22445–22451, Nov. 2019.

[15] D. B. Brückner, A. Fink, C. Schreiber, P. J. F. Röttgermann, J. O. Rädler, and C. P. Broedersz, "Stochastic nonlinear dynamics of confined cell migration in two-state systems," *Nature Physics*, vol. 15, pp. 595–601, June 2019.

[16] D. B. Brückner, P. Ronceray, and C. P. Broedersz, "Inferring the Dynamics of Underdamped Stochastic Systems," *Physical Review Letters*, vol. 125, p. 058103, July 2020.

[17] D. B. Brückner, N. Arlt, A. Fink, P. Ronceray, J. O. Rädler, and C. P. Broedersz, "Learning the dynamics of cell–cell interactions in confined cell migration," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 118, p. e2016602118, Feb. 2021.

[18] F. J. H. Heras, F. Romero-Ferrero, R. C. Hinz, and G. G. de Polavieja, "Deep attention networks reveal the rules of collective motion in zebrafish," *PLoS Computational Biology*, vol. 15, p. e1007354, Sept. 2019.

[19] M. Ruiz-Garcia, C. M. Barriuso G., L. C. Alexander, D. G. A. L. Aarts, L. M. Ghiringhelli, and C. Valeriani, "Discovering dynamic laws from observations: The case of self-propelled, interacting colloids," *Physical Review E*, vol. 109, p. 064611, June 2024.

[20] H. Koyama, H. Okumura, A. M. Ito, K. Nakamura, T. Otani, K. Kato, and T. Fujimori, "Effective mechanical potential of cell–cell interaction explains three-dimensional morphologies during early embryogenesis," *PLoS Computational Biology*, vol. 19, p. e1011306, Aug. 2023.

[21] S. Greydanus, M. Dzamba, and J. Yosinski, "Hamiltonian Neural Networks," *Advances in Neural Information Processing Systems*, vol. 32, pp. 1–16, June 2019.

[22] M. Cranmer, S. Greydanus, S. Hoyer, P. Battaglia, D. Spergel, and S. Ho, "Lagrangian neural networks," *ICLR 2020 Workshop on Integration of Deep Neural Models and Differential Equations*, Apr. 2019.

[23] K. Linka, A. Schäfer, X. Meng, Z. Zou, G. E. Karniadakis, and E. Kuhl, "Bayesian Physics Informed Neural Networks for real-world nonlinear dynamical systems," *Computer Methods in Applied Mechanics and Engineering*, vol. 402, p. 115346, Dec. 2022.

[24] L.-P. Xhonneux, M. Qu, and J. Tang, "Continuous graph neural networks," *Proceedings of the 37th International Conference on Machine Learning*, vol. 119, pp. 10432–10441, July 2020.

[25] M. Poli, S. Massaroli, J. Park, A. Yamashita, H. Asama, and J. Park, "Graph Neural Ordinary Differential Equations," *arXiv:1911.07532*, June 2021.

[26] L. Chen, K. Wu, J. Lou, and J. Liu, "Signed Graph Neural Ordinary Differential Equation for Modeling Continuous-Time Dynamics," *AAAI*, vol. 38, pp. 8292–8301, Mar. 2024.

[27] A. Han, D. Shi, L. Lin, and J. Gao, "From Continuous Dynamics to Graph Neural Networks: Neural Diffusion and Beyond," *arXiv:2310.10121*, Oct. 2023.

[28] Y. Zhang, S. Gao, J. Pei, and H. Huang, "Improving Social Network Embedding via New Second-Order Continuous Graph Neural Networks," *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 2515–2523, Aug. 2022.

[29] L. Pan, J. Lu, and X. Tang, "Spatial-temporal graph neural ODE networks for skeleton-based action recognition," *Scientific Reports*, vol. 14, p. 7629, Apr. 2024.

[30] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud, "Neural ordinary differential equations," *Advances in neural information processing systems*, vol. 31, Dec. 2018.

[31] M. Poli, S. Massaroli, A. Yamashita, H. Asama, J. Park, and S. Ermon, "Torchdyn: implicit models and neural numerical methods in pytorch," *Neural Information Processing Systems, Workshop on Physical Reasoning and Inductive Biases for the Real World*, vol. 2, Dec. 2021.

[32] X. Li, T.-K. L. Wong, R. T. Q. Chen, and D. Duvenaud, "Scalable gradients for stochastic differential equations," *International Conference on Artificial Intelligence and Statistics*, Aug. 2020.

[33] P. Kidger, J. Foster, X. Li, H. Oberhauser, and T. Lyons, "Neural SDEs as Infinite-Dimensional GANs," *International Conference on Machine Learning*, July 2021.

[34] Y. You, J. Li, S. Reddi, J. Hseu, S. Kumar, S. Bhojanapalli, X. Song, J. Demmel, K. Keutzer, and C.-J. Hsieh, "Large batch optimization for deep learning: Training BERT in 76 minutes," *International Conference on Learning Representations*, Apr. 2020.

[35] T. Hiraiwa, "Dynamic Self-Organization of Idealized Migrating Cells by Contact Communication," *Physical Review Letters*, vol. 125, no. 26, p. 268104, 2020.

[36] B. Liebchen and H. Löwen, "Modeling chemotaxis of microswimmers: From individual to collective behavior," in *Chemical Kinetics: Beyond the Textbook,* (K. Lindenberg, R. Metzler, and G. Oshanin, eds.), p. 493–516, London, United Kingdom: World Scientific Publishing Europe, 2019.

[37] M. Wang, D. Zheng, Z. Ye, Q. Gan, M. Li, X. Song, J. Zhou, C. Ma, L. Yu, Y. Gai, T. Xiao, T. He, G. Karypis, J. Li, and Z. Zhang, "Deep Graph Library: A Graph-Centric, Highly-Performant Package for Graph Neural Networks," *arXiv:1909.01315*, Aug. 2019.

[38] M. Novik, "torch-optimizer – collection of optimization algorithms for PyTorch.," Jan. 2020.

A   Supplemental Material

Computational Resources

Detailed specifications of the computational setup are provided here to ensure reproducibility. We employed the Dell Precision 7920 Tower, which comprises of 64 GB RAM, 2 Intel Xeon CPUs, and 2 NVIDIA RTX A6000 GPUs. The system operates under Linux Ubuntu 22.04, with all neural network training processes managed through pyenv with python 3.8.13 to maintain environment consistency. To use the GPUs in the computation, we used CUDA 11.5 and PyTorch 1.13.1.

Existing Assets

To construct our framework, we utilized the Deep Graph Library 1.1.1 (`https://www.dgl.ai/`) (37), TorchSDE 0.2.5 (`https://github.com/google-research/torchsde`) (32; 33), TorchDyn 1.0.4 (`https://github.com/DiffEqML/torchdyn`) (31), and torch-optimizer 0.3.0 (`https://github.com/jettify/pytorch-optimizer`) (38), all of which are publicly available under the MIT License and Apache License 2.0.

Supplemental Tables

Trial ID	γ	$M_{\mathbf{tra}}$	$\mathbf{MSE}(F_v^{(1)})$	$\mathbf{MAE}(F_v^{(1)})$	$\mathbf{MSE}(F_v^{(2)})$	$\mathbf{MAE}(F_v^{(2)})$
1	1×10^{-2}	270	3.6×10^{-3}	5.9×10^{-2}	5.1×10^{-3}	4.8×10^{-2}
2	1×10^{-2}	3	1.7	1.4	2.2×10^{-2}	1.4×10^{-1}
3	1×10^{-2}	3	9.4	3.7	2.2×10^{-2}	1.3×10^{-1}
4	1×10^{-1}	3	1.2×10^{-2}	1.7×10^{-2}	2.2×10^{-2}	1.0×10^{-1}
5	1×10^{-1}	3	5.4×10^{-2}	3.8×10^{-2}	2.3×10^{-2}	1.1×10^{-1}

Supplemental Table S1: Normalized MSE and MAE of the estimation of $F^{(1)}$ and $F^{(2)}$ for the harmonic interaction model. Each row indicates an independent experiment.

Parameter set	$\alpha_{CF}(0)$	$\alpha_{CF}(1)$	$\alpha_{Ch}(0)$	$\alpha_{Ch}(1)$
(i)	0.1	0.9	2.0	0.2
(ii)	0.9	0.5	0.5	0.5
(iii)	0.9	0.9	0.2	0.2
(iv)	0.9	0.9	0.5	0.5
(v)	0.1	0.1	2.0	2.0

Supplemental Table S2: Parameter sets used in the simulations of the mixed-species model.

Trial ID	Parameter set	$\mathbf{MSE}(F_r^{(1)})$	$\mathbf{MAE}(F_r^{(1)})$	$\mathbf{MSE}(F_\phi^{(1)})$	$\mathbf{MAE}(F_\phi^{(1)})$	$\mathbf{MSE}(F_r^{(2)})$	$\mathbf{MAE}(F_r^{(2)})$	$\mathbf{MSE}(F_\phi^{(2)})$	$\mathbf{MAE}(F_\phi^{(2)})$
1	(i)	1.5×10^{-3}	3.7×10^{-2}	4.5×10^{-7}	6.7×10^{-4}	5.6×10^{-1}	2.7×10^{-1}	2.4×10^{-1}	2.8×10^{-1}
2	(i)	1.7×10^{-3}	3.5×10^{-2}	3.0×10^{-7}	3.9×10^{-3}	9.9×10^{-1}	3.0×10^{-1}	1.6×10^{-1}	2.8×10^{-1}
3	(ii)	2.0×10^{-3}	4.2×10^{-2}	3.8×10^{-6}	1.7×10^{-3}	1.9×10^{-1}	1.6×10^{-1}	5.4×10^{-2}	2.0×10^{-1}
4	(ii)	5.8×10^{-4}	2.3×10^{-2}	7.1×10^{-6}	2.7×10^{-3}	2.7×10^{-1}	1.7×10^{-1}	5.8×10^{-2}	2.1×10^{-1}
5	(iii)	1.0×10^{-3}	3.1×10^{-2}	2.3×10^{-6}	1.1×10^{-3}	4.2×10^{-1}	1.9×10^{-1}	6.8×10^{-2}	2.5×10^{-1}
6	(iv)	1.6×10^{-4}	1.3×10^{-2}	3.2×10^{-5}	5.0×10^{-3}	3.7×10^{-1}	1.7×10^{-1}	7.3×10^{-2}	2.4×10^{-1}
7	(v)	2.3×10^{-3}	4.8×10^{-2}	3.9×10^{-6}	1.9×10^{-3}	2.8×10^{-1}	1.9×10^{-1}	8.8×10^{-2}	1.6×10^{-1}
8	(v)	6.4×10^{-3}	8.0×10^{-2}	3.1×10^{-5}	5.4×10^{-3}	3.9×10^{-1}	2.3×10^{-1}	1.4×10^{-1}	2.0×10^{-1}

Supplemental Table S3: Normalized MSE and MAE of the estimation of $F^{(1)}$ and $F^{(2)}$ for the mixed-species model. Each row indicates an independent experiment. To note, $\mathbf{MSE}(F_\phi^{(1)})$ and $\mathbf{MAE}(F_\phi^{(1)})$ are not normalized since the true value is zero in any case.

Supplemental figures
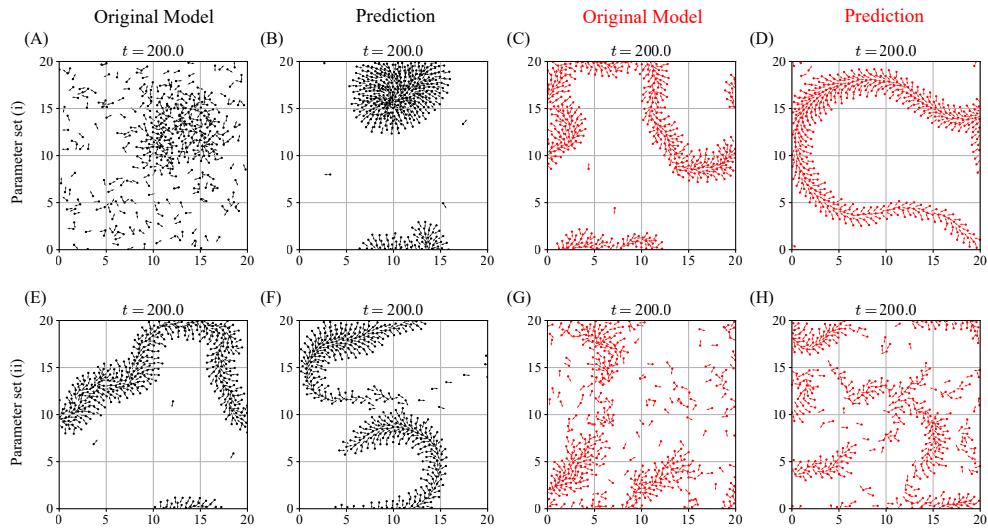
Supplemental Figure S1: Snapshots of the simulation results in the mixed-species model and the estimated model. Panels (A, C, E, G) depict results from the mixed-species model, and panels (B, D, F, H) from the estimated model. The upper rows (A-D) represent simulations with parameter set (i), and the lower rows (E-H) with parameter set (ii). Panels (A, B, E, F) include individuals with $c^i = 0$ and panels (C, D, G, H) with $c^i = 1$. Colors correspond to those used in Figure 3.
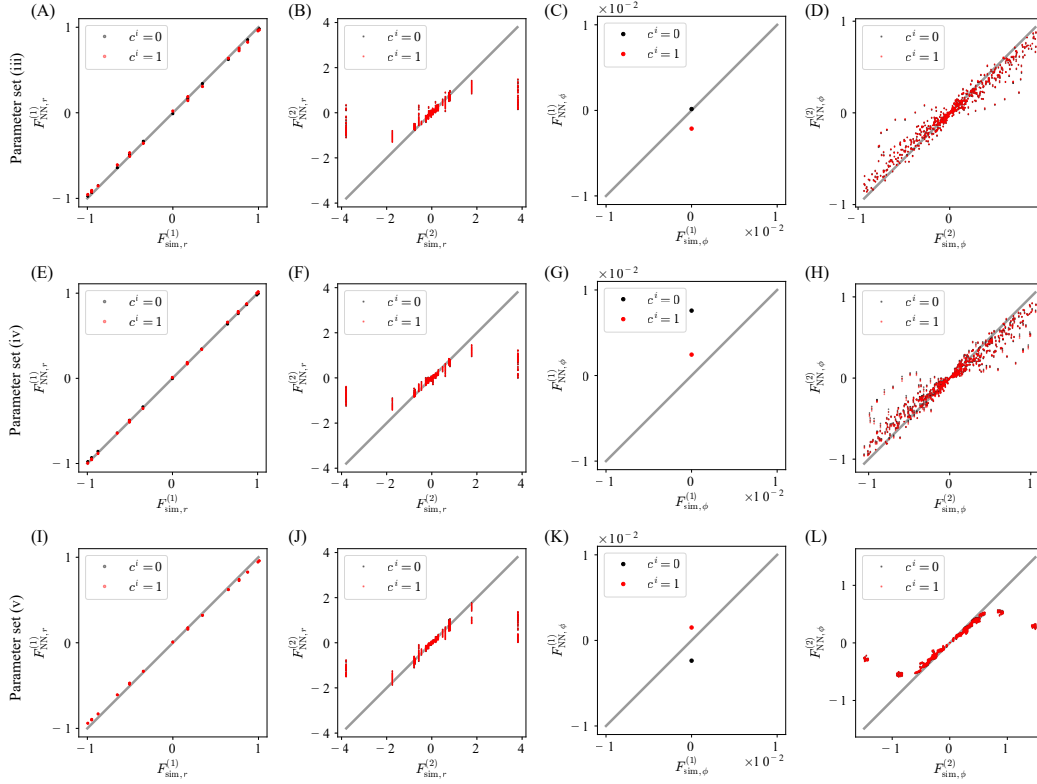
Supplemental Figure S2: The functions estimated from data with the mixed-species model, without dependency on species type, plotted against the true values. The rows display different cases: (A-D) with parameter set (iii), (E-H) with parameter set (iv), and (I-L) with parameter set (v). A gray line serves as a guide to indicate where the estimated values equal the true values.

Supplemental Movies

Supplemental Movie S1:

A visualization of the simulation based on the harmonic interaction model with friction constant $\gamma = 1 \times 10^{-2}$.

Supplemental Movie S2:

A visualization of the simulation based on the harmonic interaction model with friction constant $\gamma = 1 \times 10^{-1}$.

Supplemental Movie S3:

The results of simulations using the estimated functions $F_{NN}^{(1)}$ and $F_{NN}^{(2)}$, which were trained using $N_{tra} = 270$ data from the harmonic interaction model with $\gamma = 1 \times 10^{-2}$.

Supplemental Movie S4:

The results of simulations using the estimated functions $F_{NN}^{(1)}$ and $F_{NN}^{(2)}$, which were trained using $N_{tra} = 3$ data from the harmonic interaction model with $\gamma = 1 \times 10^{-2}$.

Supplemental Movie S5:

The results of simulations using the estimated functions $F_{NN}^{(1)}$ and $F_{NN}^{(2)}$, which were trained using $N_{tra} = 3$ data from the harmonic interaction model with $\gamma = 1 \times 10^{-1}$.

Supplemental Movie S6:

A visualization of the simulation based on the mixed-species model with parameter set (i).

Supplemental Movie S7:

A visualization of the simulation based on the mixed-species model with parameter set (ii).

Supplemental Movie S8:

The results of simulations using the estimated functions $F_{\mathrm{NN}}^{(1)}$ and $F_{\mathrm{NN}}^{(2)}$, which were trained using $N_{\mathrm{tra}} = 3$ data from the mixed-species model with parameter set (i).

Supplemental Movie S9:

The results of simulations using the estimated functions $F_{\mathrm{NN}}^{(1)}$ and $F_{\mathrm{NN}}^{(2)}$, which were trained using $N_{\mathrm{tra}} = 3$ data from the mixed-species model with parameter set (ii).

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: See Section 1.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See Section 6.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: The paper do not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: See Section 4-5.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: See the Supplemental Material.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: See Section 4-5.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: Due to the extensive computation time required for estimation, comprehensive statistics could not be gathered. Instead, we present the results of all trials to illustrate the trends in estimation accuracy (see Supplemental Tables 1,3).

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: See the Supplemental Material.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [No]

Justification:

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [No]

Justification:

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: See the Supplemental Material

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: See the Supplemental Materials.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.