NEURAL INFORMATION PROCESSING SYSTEMS

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

THE UNIVERSITY OF TOKYO

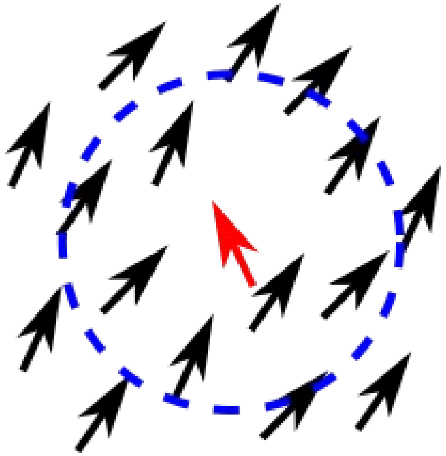
生物普遍性研究機構
Universal Biology Institute

Collective motion of active matter

- **Active matter**:
  Can self-propel by consuming energy in environment

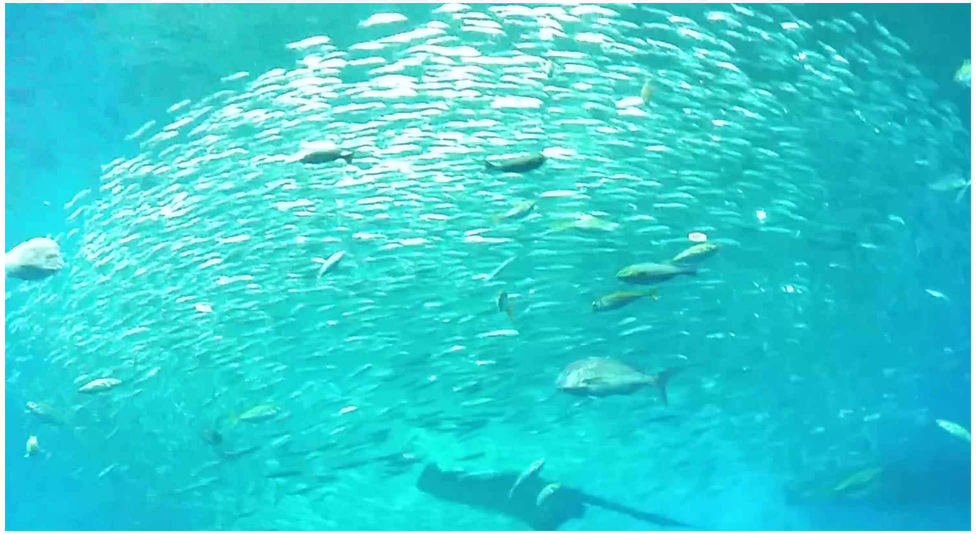
- **Collective motion**:
  Group of active matter entities shows self-organized behavior through interaction

- In active matter physics, collective motion is often modeled by position and polarity of entities



Vicsek, T. *et al.* *PRL* (1995).

Fish School



Bird Flock

Cavagna, A. *et al.* *PNAS* (2010).

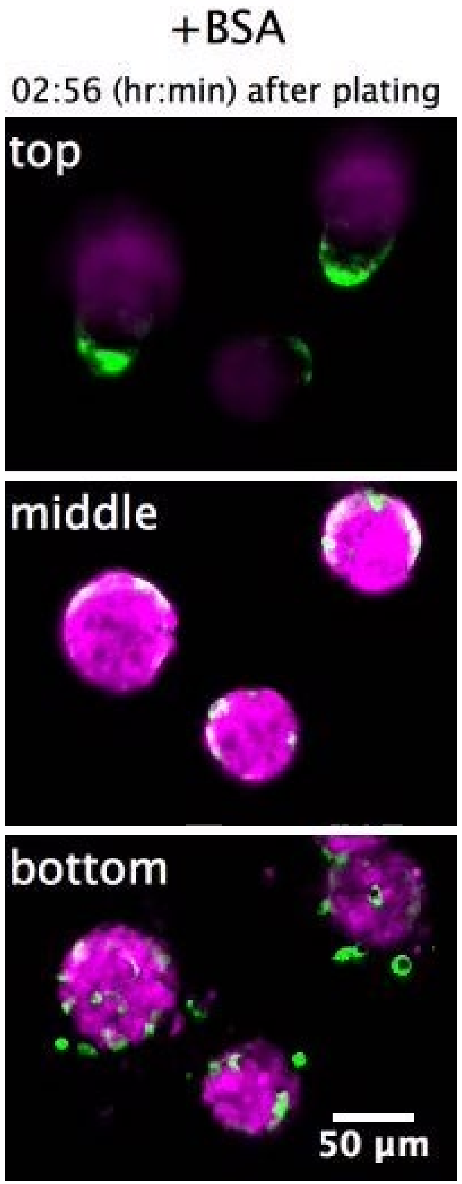
Non-reciprocal mixed-species collective motion

- **Mixed-species**:
  Morphogenesis requires variable cell types
  to make different organs

- **Non-reciprocal**:
  Interactions can affect internal states such as polarity
  → can be non-reciprocal

Morphogenesis of *Dictyostelium discoideum*

Fujimori, T., *et al. PNAS* (2019).

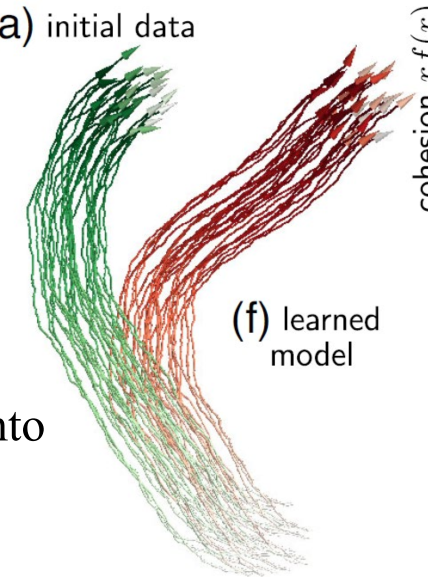


Estimating the rules for multi-body dynamics

- Existing models modeled self-propulsion and interaction in several ways

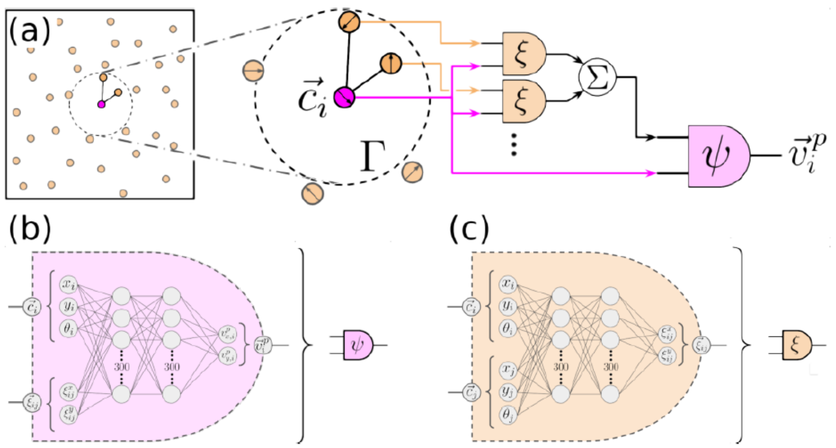
- Rules are estimated by minimizing the predicting error of trajectories

- Not applied to mixed-species systems



Decompose forces into basal function series

Brückner, D. B. *et al. PRL* (2020).



Velocity is predicted by deep neural network that receive a pair of polarities

Ruiz-Garcia, M. *et al. PRE* (2024).

Our framework

- General form of equation of motion in multi-body systems

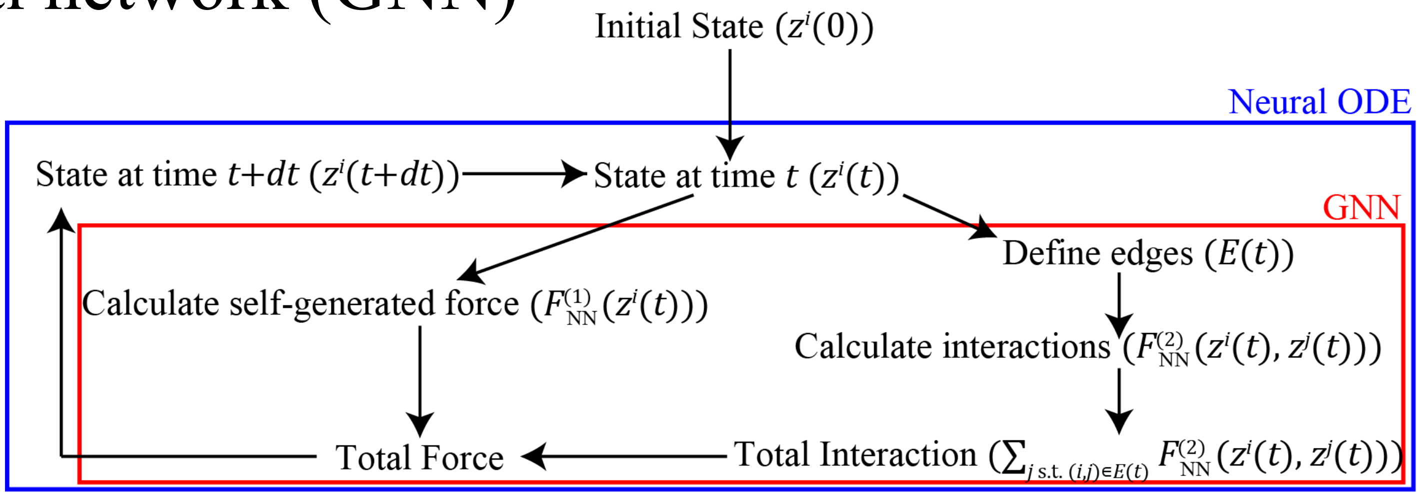
$z^i(t) = (x^i(t), y^i)$ Dynamic (x) and static (y) variables

- Truncated at pairwise interaction

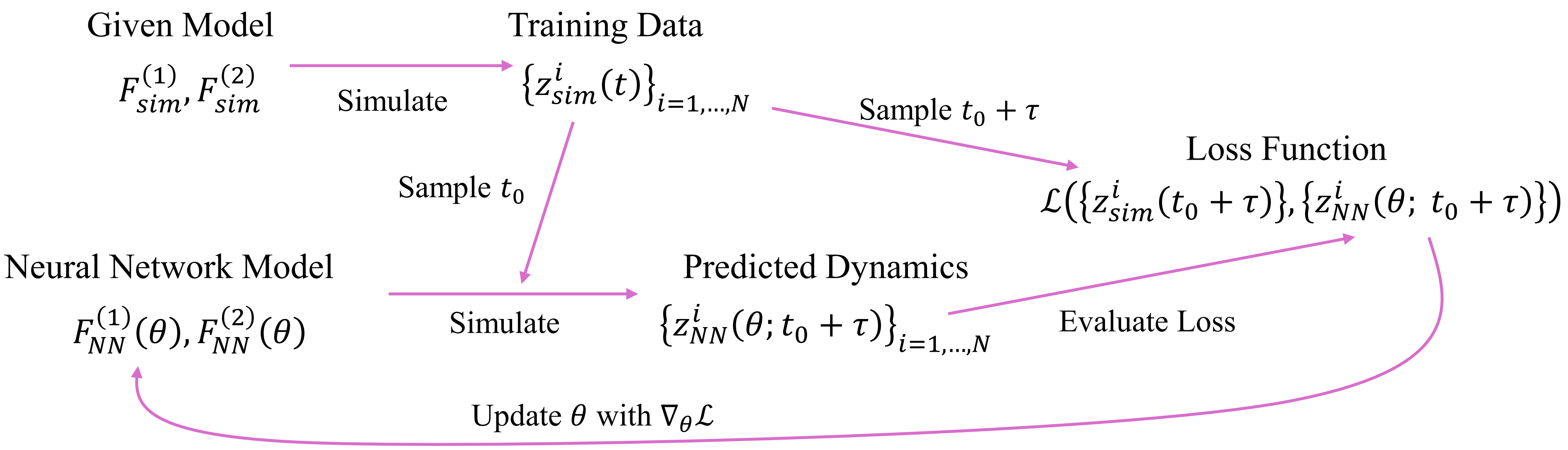
$$dx^i = \left(F^{(1)}(z^i(t)) + \sum_{j \text{ s.t. } d_{ij} < d_0} F^{(2)}(z^i(t), z^j(t)) \right) dt + \sigma dW^i(t)$$

- Neural ODE calls Graph neural network (GNN) at each calculation step
- GNN updates edges with pre-defined rule and returns total force (improved from GraphODE)

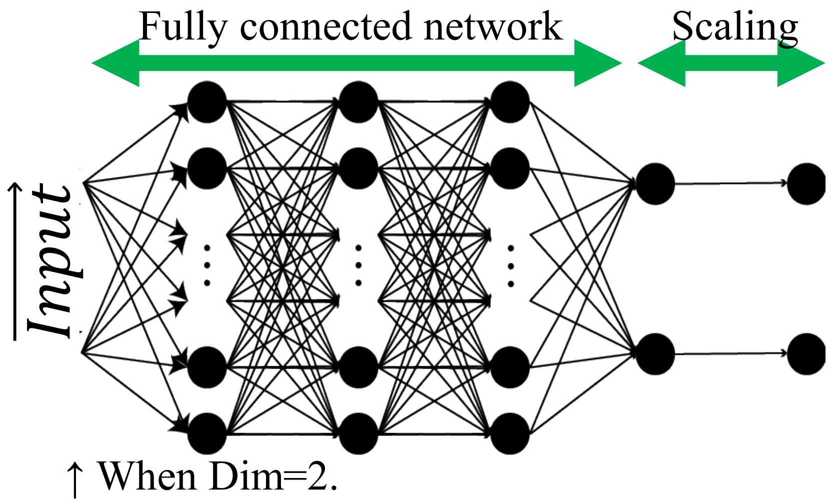
Poli, M. *et al.* *arXiv* (2021).



Method for estimation



- Neural networks are trained to predict the trajectories in given model



<u>Fully connected network</u>

Depth: 3

Width: 128

Activation: ELU

$$ELU(x) = \max(x, e^x - 1)$$

<u>Scaling Layer</u>

$$Scaling(\vec{x}) = e^{A}\vec{x} + \vec{B}$$

Initial: $A = -20,\ \vec{B} = \vec{0}$

<u>Optimizer</u> : LAMB

Training data 1

Movies →

- Underdamped Brownian Motion with Harmonic Interaction

$x^i = (r^i, v^i)$ Position, Velocity

$y^i = ()$ None

$$dr^i = v^i dt,$$

$$dv^i = \Big(\underbrace{-\gamma v^i}_{\text{friction}} - \sum_{j \text{ s.t.} (i,j) \in E(t)} \underbrace{\nabla_{r^i} U(r^i - r^j)}_{\text{harmonic interaction}} \Big) dt + \sigma dW^i(t)$$

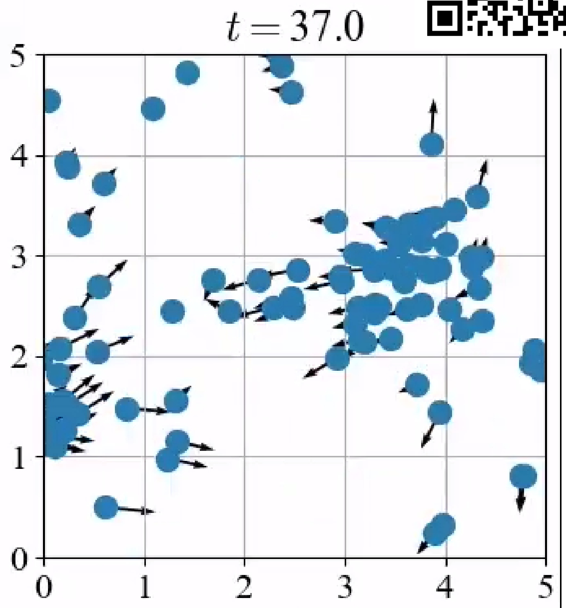
$$U(r) = \frac{1}{2} k (|r| - r_c)^2$$

$$dr^i = v^i dt,$$

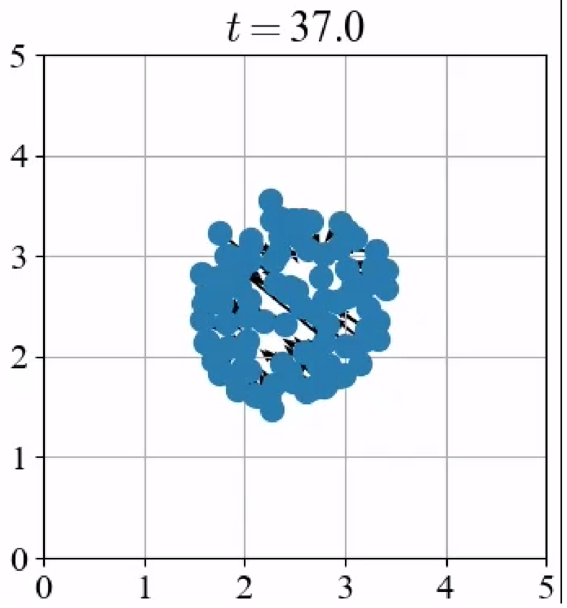
$$dv^i = \left(F^{(1)}_{NN,v}(z^i; \theta) + \sum_{j \text{ s.t.} (i,j) \in E(t)} F^{(2)}_{NN,v}(z^i, z^j; \theta) \right) dt$$

→ Trained for loss function (normalized prediction error of (r^i, v^i))

Weak friction
$\gamma = 1 \times 10^{-2}$

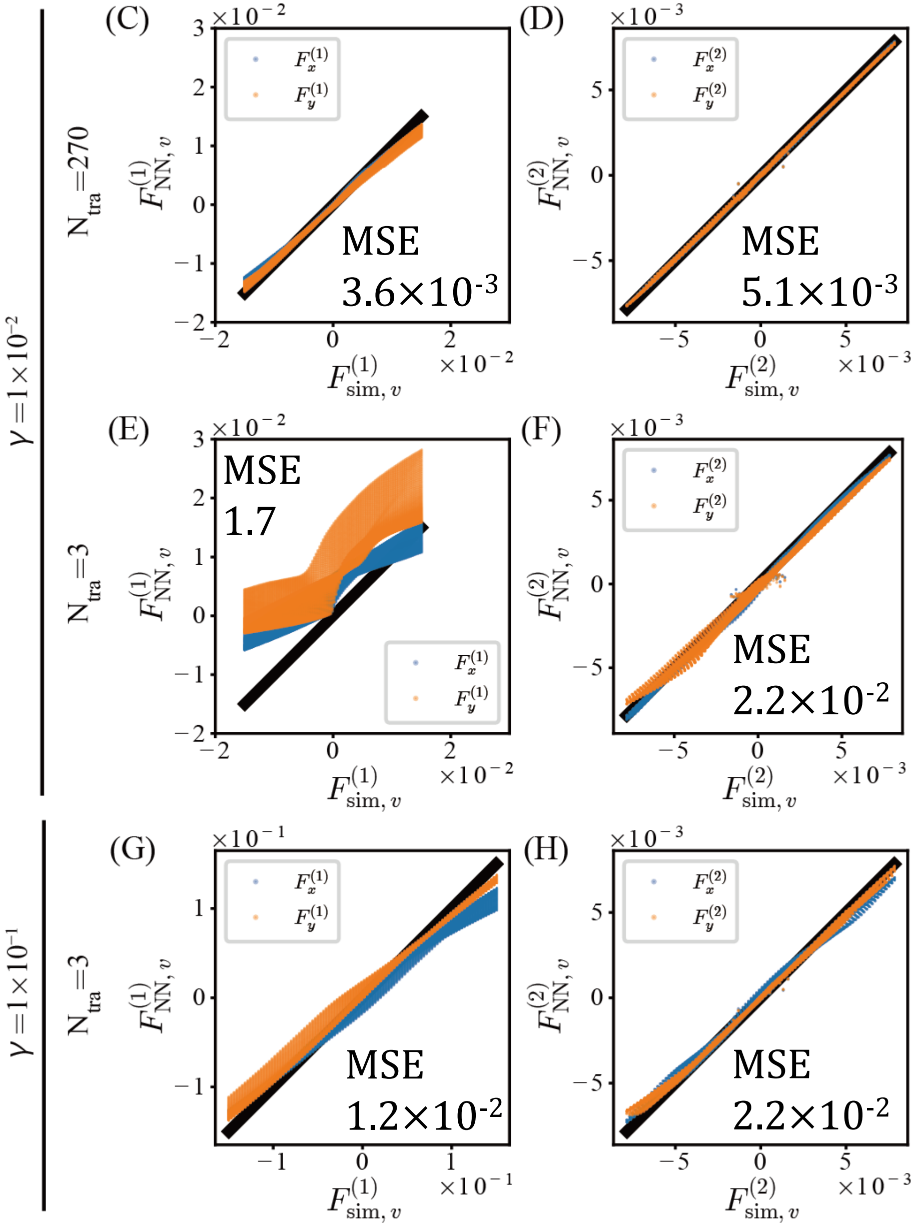


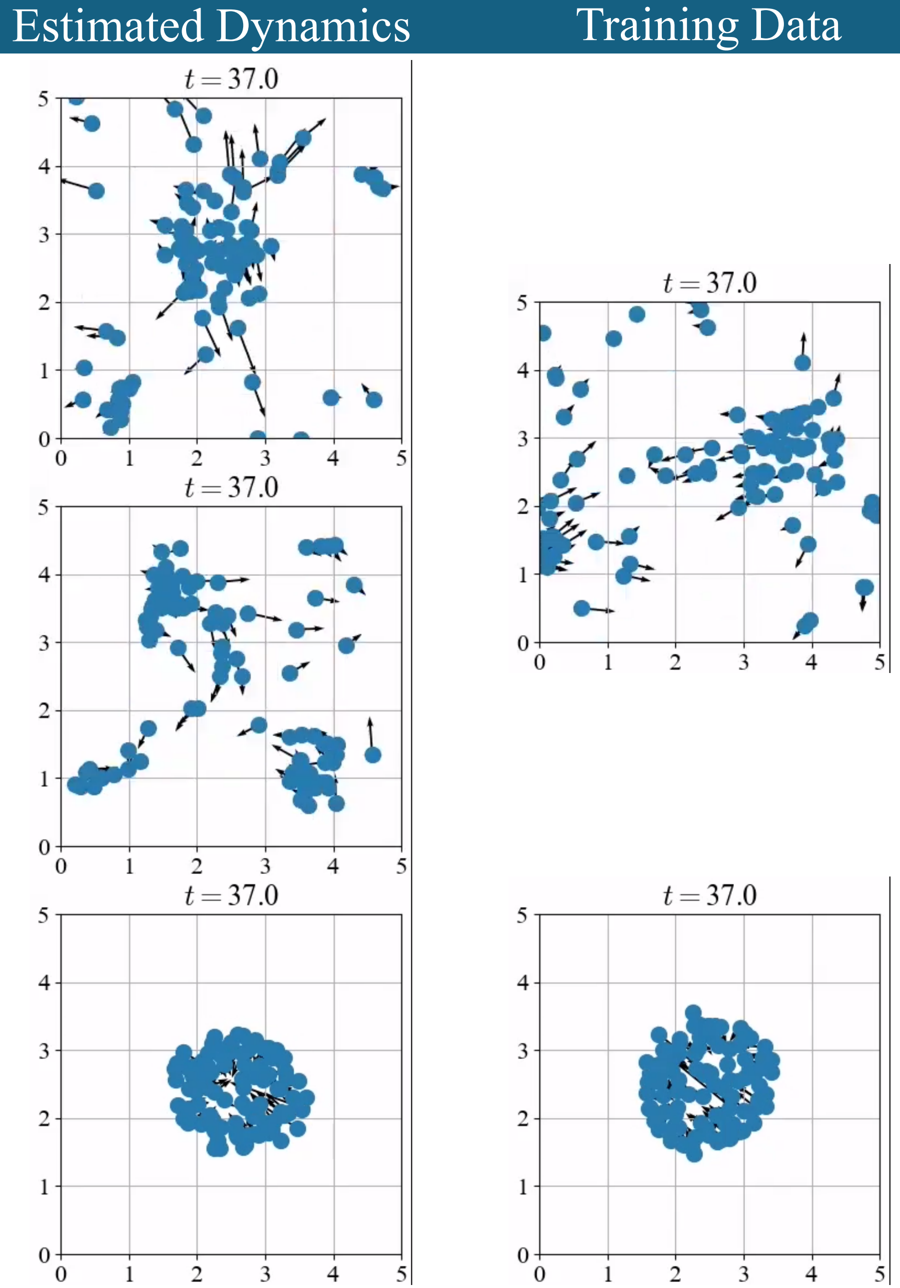
Strong friction
$\gamma = 1 \times 10^{-1}$



Estimation for training data 1

Movies →





Training data 2

Movies →

- Mixed Species Collective Motion with Overdamped Self-propulsion

$x^i = (r^i, \phi^i)$ Position, Polarity angle

$y^i = (c^i \in \{0,1\})$ Species type

$$dr^i = (v_0 p^i + \sum_{j \text{s.t.} (i,j)\in E(t)} \underbrace{\beta J_{\text{eV}}^{ij}}_{\text{exclusion volume}})dt$$

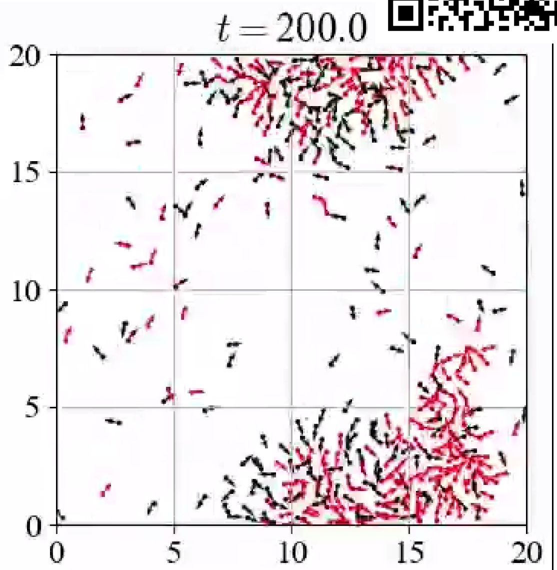
$$d\phi^i = -\sum_{j \text{s.t.} (i,j)\in E(t)} \left(\underbrace{\alpha_{\text{CF}}(c^i) J_{\text{CF}}^{ij}}_{\text{contact following}^1} + \underbrace{\alpha_{\text{Ch}}(c^i) J_{\text{Ch}}^{ij}}_{\text{chemotaxis}^2} \right) (r^{ij} \cdot p_\perp^i)\, dt + \sigma dW^i(t).$$

$$J_{\text{eV}}^{ij} = (r_c^{-1} - |r^{ij}|^{-1})\, r^{ij}, \qquad p^i = (\cos\phi^i, \sin\phi^i), p_\perp^i = (-\sin\phi^i, \cos\phi^i)$$

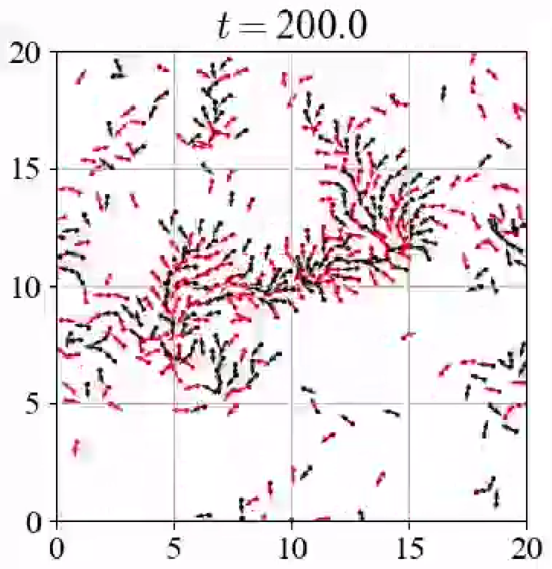
$$J_{\text{CF}}^{ij} = \frac{1}{2}\left(1 - \frac{r^{ij}\cdot p^j}{|r^{ij}|}\right), \qquad r^{ij} = r^j - r^i \in [-L/2, L/2]^2.$$

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

$\alpha_{CF}(0) = 0.1$
$\alpha_{CF}(1) = 0.9$
$\alpha_{Ch}(0) = 2.0$
$\alpha_{Ch}(1) = 0.2$



$\alpha_{CF}(0) = 0.9$
$\alpha_{CF}(1) = 0.5$
$\alpha_{Ch}(0) = 0.5$
$\alpha_{Ch}(1) = 0.5$



[1]Hiraiwa, T. *PRL* (2020). [2]Liebchen, B. & Löwen, H. *Chemical kinetics: Beyond the textbook,* 493–516 (2019).

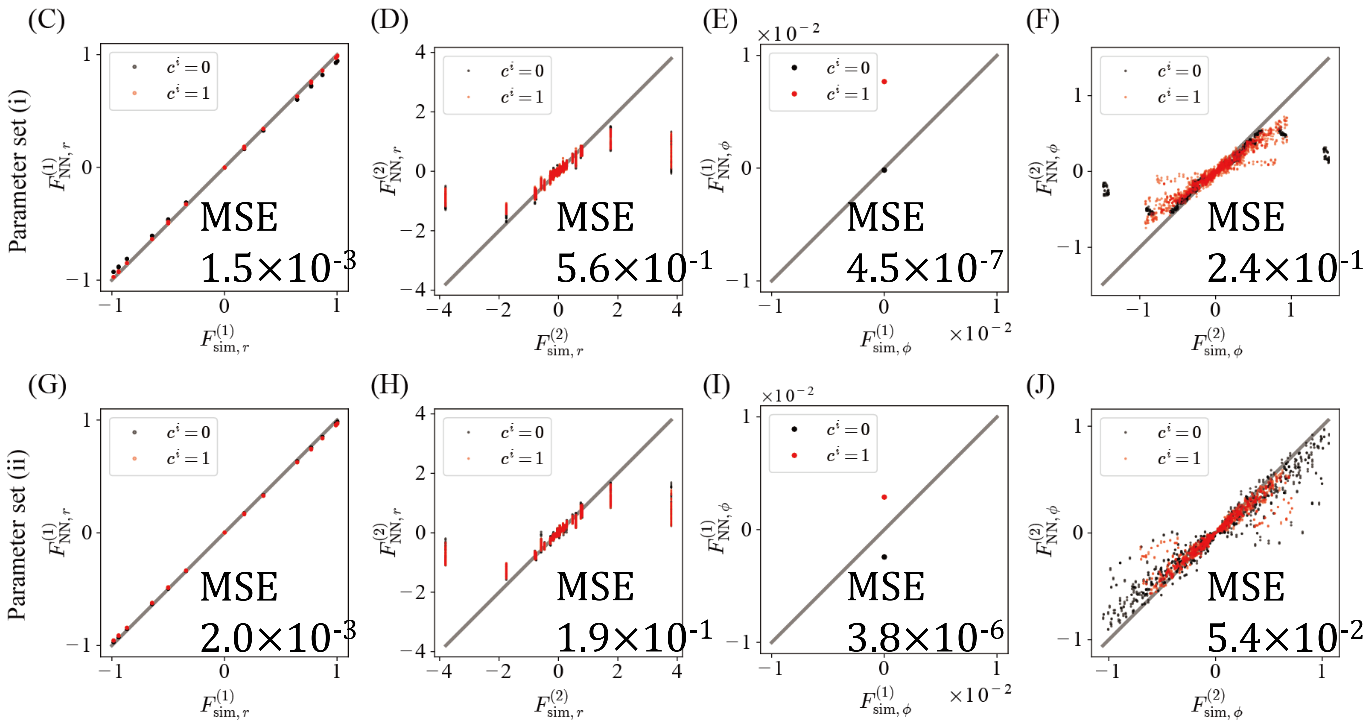
Estimation for training data 2

Movies →

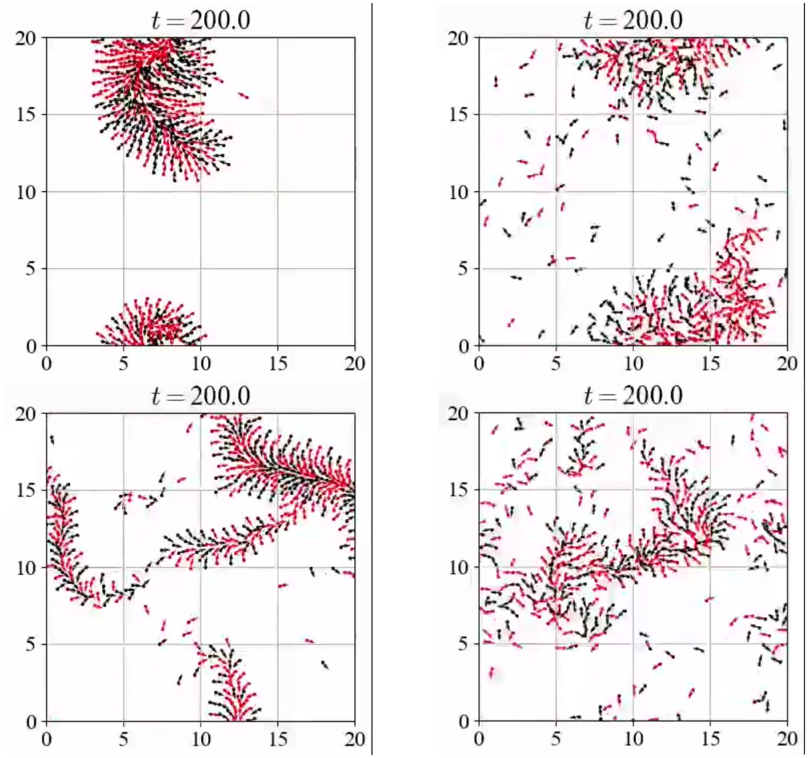
$$dr^i = \left(F^{(1)}_{NN,r}(z^i;\theta) + \sum_{j\ s.t.(i,j)\in E(t)} F^{(2)}_{NN,r}(z^i, z^j;\theta) \right) dt,$$

$$d\phi^i = \left(F^{(1)}_{NN,\phi}(z^i;\theta) + \sum_{j\ s.t.(i,j)\in E(t)} F^{(2)}_{NN,\phi}(z^i, z^j;\theta) \right) dt$$

→ Trained for loss function
(normalized prediction error of (r^i, ϕ^i))



Estimated Dynamics　　Training Data

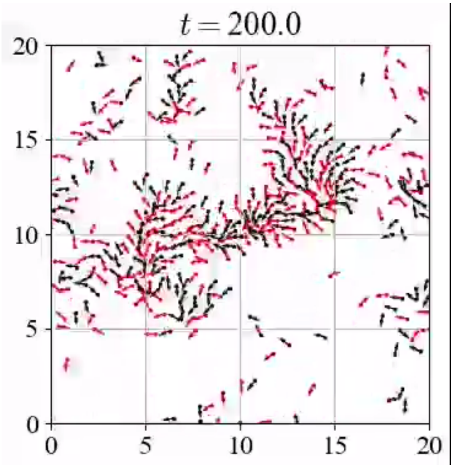


Conclusion

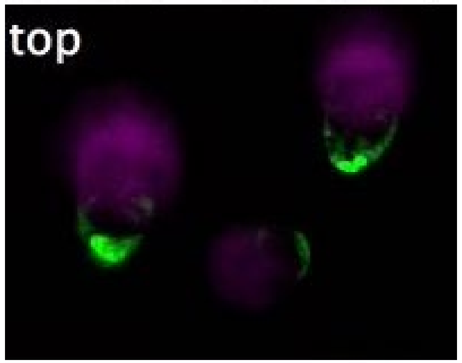
- GNN + neuralODE can learn forces from trajectories

Dataset

Numerical Simulation



Experimental Data



Fujimori, T. *et al.* *PNAS* (2019).

Proposed Framework

Estimated Equation of motion

Self-propulsion,
Spring-like interaction,
Contact following
+ Chemotaxis

Known- or Unknown-
Self-propulsion,
Interaction

Paper

Project Page

Thank you!