Behavior-Inspired Neural Networks for Relational Inference

Yulong Yang[*,†] **Bowen Feng**[*] **Keqin Wang**
Naomi Ehrich Leonard **Adji Bousso Dieng** **Christine Allen-Blanchette**[†]
Princeton University

Abstract

From pedestrians to Kuramoto oscillators, interactions between agents govern how dynamical systems evolve in space and time. Discovering how these agents relate to each other has the potential to improve our understanding of the often complex dynamics that underlie these systems. Recent works learn to categorize relationships between agents based on observations of their physical behavior. These approaches model relationship categories as outcomes of a categorical distribution which is limiting and contrary to real-world systems, where relationship categories often intermingle and interact. In this work, we introduce a level of abstraction between the observable behavior of agents and the latent categories that determine their behavior. To do this, we learn a mapping from agent observations to agent preferences for a set of latent categories. The learned preferences and inter-agent proximity are integrated in a nonlinear opinion dynamics model, which allows us to naturally identify mutually exclusive categories, predict an agent's evolution in time, and control an agent's behavior. Through extensive experiments, we demonstrate the utility of our model for learning interpretable categories, and the efficacy of our model for long-horizon trajectory prediction.

1 Introduction

Multi-agent systems are found in domains as diverse as astronomy (Villanueva-Domingo et al., 2022; Lemos et al., 2023), biology (Fiorelli et al., 2006; Seeley et al., 2012; Young et al., 2013), physics (Gull et al., 2013; Browaeys & Lahaye, 2020), and sports (Hauri et al., 2021; Wang et al., 2024). Understanding how these systems evolve in time has the potential to provide insights useful for the discovery of unknown physics, the rules governing collective behavior, and engineering design. Predicting the evolution of complex systems is a fundamental challenge in the learning literature. Early black-box models determine future states from past states without regard for contextual information (Hochreiter & Schmidhuber, 1997; Sutskever et al., 2014). More recent approaches improve upon these models by incorporating information about environmental conditions and the behavior of other agents (Alahi et al., 2016; Gupta et al., 2018; Casas et al., 2018; Deo & Trivedi, 2018; Sadeghian et al., 2019). While incorporating contextual information has led to more powerful trajectory prediction models, the opacity of these models limits our ability to leverage them to better understand the role of inter-agent relationships.

Recent work in relational inference attempts to address this limitation by explicitly modeling inter-agent relationships (Sukhbaatar et al., 2016; Santoro et al., 2017; Kipf et al., 2018; Graber & Schwing, 2020; Xu et al., 2022, 2023). Discovering inter-agent relationships is challenging, however, since ground truth labels are typically unavailable, and the relevant relationships may be unknown at design time. Graph neural network (GNN) based approaches such as Kipf et al. (2018); Graber & Schwing (2020); Xu et al. (2022, 2023) model inter-agent relationship categories with a categorical variable. In their seminal work, the authors of Kipf et al. (2018) learn a latent representation of inter-agent relationships in a trajectory prediction pipeline. The authors of Graber & Schwing (2020) improve on this approach by modeling inter-agent relationships as time varying and the authors of Xu et al. (2022) improve model expressivity by representing node features as a hypergraph. While these methods are able to predict future observations of a number of systems, the underlying assumption that inter-agent relationships are determined by a single relationship category diverges from what we observe in the real world.

Proceedings of the 28th International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258. Copyright 2025 by the author(s). [*]Equal contribution. [†]Corresponding author.

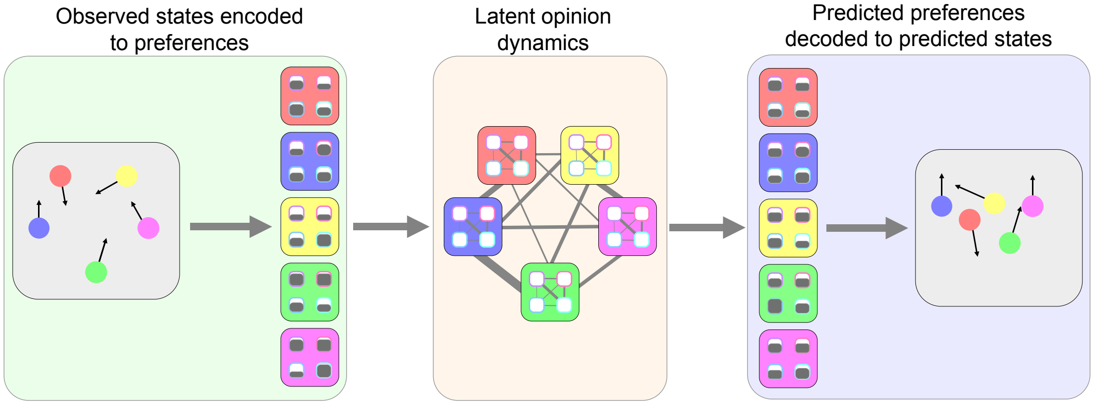

Observed states encoded to preferences
Latent opinion dynamics
Predicted preferences decoded to predicted states

Figure 1: **Behavior-inspired neural network for relational inference.** BINNs allow for relational inference using inter-agent distance and a learned representation of inter-category interactions. The physical states of agents are encoded to preferences for a set of latent categories. Preferences are propagated forward in time using a nonlinear opinion dynamics model with learned parameters. Predicted preferences are then decoded to physical states as predicted future physical states.

In contrast to these models, opinion dynamics models assume agents have preferences for multiple categories, and that these preferences evolve in time. The nonlinear opinion dynamics (NOD) model introduced in Leonard et al. (2024), has been used to model a variety of societal systems (Das et al., 2014; Rossi & Frasca, 2020; Leonard et al., 2021; Franci et al., 2021; Cathcart et al., 2022; Bizyaeva et al., 2022a, 2024; Hu et al., 2023; Ordorica Arango et al., 2024; Wang et al., 2025). The nonlinear nature of this model introduces a bifurcation which allows preferences to quickly and flexibly respond to changes in environmental inputs. A limitation of this method, however, is that the relationship between agent preferences for a set of categories and their observable behavior must be known a priori.

In our model, *Behavior-Inspired Neural Network* (BINN), we combine the flexibility of GNN models with the interpretability of nonlinear opinion dynamics for a new approach to relational inference. Concretely, our contributions are the following:

- In contrast to existing opinion dynamics models which require observability of agent preferences, we learn a representation of agent preferences from physical observations.
- In contrast to existing relational inference approaches which model relationship categories as a categorical variable, we use the NOD inductive bias to model the evolution of real-valued agent preferences on a set of latent categories.
- By incorporating the NOD inductive bias, we can identify mutually exclusive categories, predict an agent's evolution in time, and control an agent's behavior.

We demonstrate the utility of our approach for identifying mutually exclusive categories on multiple illustrative examples and demonstrate the efficacy of our approach for trajectory prediction tasks.

2 Related Work

Trajectory prediction. In contrast to traditional control approaches which tune a set of model parameters from all available trajectories (Brunton et al., 2016; Galioto & Gorodetsky, 2020; Paredes & Bernstein, 2021, 2024; Paredes et al., 2024; Richards et al., 2024), early deep learning approaches learn to map a sequence of input states to future states directly (Sutskever et al., 2014). Later efforts incorporated external influences (e.g., behavior of other agents, environmental conditions) (Alahi et al., 2016; Gupta et al., 2018; Casas et al., 2018; Deo & Trivedi, 2018; Sadeghian et al., 2019), physical priors (Greydanus et al., 2019; Mason et al., 2022, 2023; Allen-Blanchette, 2024), and temporal dependencies (Vemula et al., 2018; Sadeghian et al., 2019; Mangalam et al., 2020; Yuan et al., 2021; Giuliari et al., 2021; Rezaei & Dieng, 2024), and recent work has used graph based methods to model multi-agent dynamics (Yu et al., 2020; Gao et al., 2020). While these methods can predict future system states, they do not predict which agents interact or how they interact, a limitation which motivates relational inference.

Relational inference. The goal of relational inference is to infer inter-agent relationships in a multi-agent system. This task is challenging since, in general, the relationships between individual agents are unobservable. Early works (Sukhbaatar et al., 2016; Foerster et al., 2016) focus on learning the communication protocols for multi-agent systems. Neural Relational Inference (Kipf et al., 2018) proposed a variational autoencoding (Kingma et al., 2013) graph neural network framework for learning time-invariant relationships between individual agents. Further developments focused on increasing the expressivity and applicability of this framework by incorporating factorized graphs to model different types of interactions (Webb et al., 2019); dynamic encoders to model time-varying inter-agent relations Graber & Schwing (2020); edge-to-edge message passing for more efficient information sharing (Chen et al., 2021); shared decoder conditioned on edge types to learn a mapping for future states (Löwe et al., 2022); hypergraphs to accommodate interactions of different spatial scales (Xu et al., 2022); and Euclidean transformation equivariance to improve generalization to varying scenes Xu et al. (2023). Other works leverage different architectures to infer inter-agent relations such as meta-learning to map inputs to edge and node values (Alet et al., 2019); reservoir computing to increase the efficiency of relational inference (Wang et al., 2023); partial correlates of latent representations to infer connections (Wang & Pang, 2024); and diffusion models to reconstruct missing connections (Zheng et al., 2024). Our work differs from these approaches in our representation of relationship categories. Our categories exist in a space of agent preferences rather than the physical space, and our categories are flexible and interacting, rather than mutually exclusive and independent.

Consensus dynamics. In control and robotics, consensus dynamics (Bullo, 2018) have been used in a myriad of settings to model the dynamics and control the behavior of multi-agent systems. In Levine et al. (2000), the authors propose a linear model for prediction and control of multi-agent systems. In Leonard et al. (2010), the authors use a linear consensus dynamics model for coordinated surveying with underwater gliders. In Justh & Krishnaprasad (2005), the authors use the consensus dynamics framework to develop rectilinear and circular consensus control laws for multi-vehicles systems. In Leonard et al. (2007), the authors use consensus dynamics to improve data collection in mobile sensor networks. In Ballerini et al. (2008), the authors use consensus dynamics to understand the robustness of starling flocking behavior. Even with this breadth of application, there are drawbacks to a linear model of opinion formation; specifically, the naive implementation results in dynamics that only yields consensus (Altafini, 2012; Dandekar et al., 2013) and the formation of opinions in response to inputs is slow.

Nonlinear opinion dynamics. The noted short comings of linear consensus dynamics models are resolved in the nonlinear opinion dynamics model proposed in Leonard et al. (2024). As the model is nonlinear, the dynamics result in a bifurcation and opinions can evolve to dissensus quickly. Nonlinear opinion dynamics have been used to model a variety of systems. In Das et al. (2014); Rossi & Frasca (2020); Leonard et al. (2021); Franci et al. (2021) nonlinear opinion dynamics model information spread in settings such as political polarisation. In Hu et al. (2023), nonlinear opinion dynamics is used to resolve deadlock, and in Cathcart et al. (2022), it is used for collision avoidance in human-robotic systems. In Bizyaeva et al. (2024); Ordorica Arango et al. (2024), nonlinear opinion dynamics is used for modeling risk in epidemic models. In contrast to these works, where the model has direct access agent opinions (i.e., preferences), we learn a mapping between physical states and agent preferences.

3 Background

Nonlinear dynamics differ from linear dynamics in that they are able to exhibit bifurcations (Golubitsky et al., 2012; Guckenheimer & Holmes, 2013; Strogatz, 2018). Nonlinear opinion dynamics (Bizyaeva et al., 2022b) leverage bifurcations for fast and flexible decision making even with weak input signals (Leonard et al., 2024). Our BINN model integrates a nonlinear opinion dynamics inductive bias to interpretably and flexibly model agent behavior.

3.1 Nonlinear opinion dynamics

Consider a multi-agent system of $\mathcal{N}_\mathrm{a} \in \mathbb{N}$ agents, each possessing real-valued preferences about $\mathcal{N}_\mathrm{o} \in \mathbb{N}$ categories. Each category represents a belief or desire (e.g., take a physical action, undertake a task), and an agent's preference for a category can be positive, neutral, or negative (the preference magnitude corresponds to preference strength). Changes to an agent's preference for a category depend on extrinsic and intrinsic parameters. Concretely, the changes to agent i's preference for category j can be determined by the differential equation proposed in Leonard et al. (2024)

$$\begin{aligned}\dot{z}_{ij} = -d_{ij} z_{ij} + \mathcal{S}\Bigg(u_i \Bigg(\alpha_{ij} z_{ij} + \sum_{\substack{k=1\\k\neq i}}^{\mathcal{N}_\mathrm{a}} a^{\mathrm{a}}_{ik} z_{kj} \\ + \sum_{\substack{l=1\\l\neq j}}^{\mathcal{N}_\mathrm{o}} a^{\mathrm{o}}_{jl} z_{il} + \sum_{\substack{k=1\\k\neq i}}^{\mathcal{N}_\mathrm{a}} \sum_{\substack{l=1\\l\neq j}}^{\mathcal{N}_\mathrm{o}} a^{\mathrm{a}}_{ik} a^{\mathrm{o}}_{jl} z_{kl} \Bigg) \Bigg) + b_{ij}. \quad (1)\end{aligned}$$

The parameters $d_{ij} \geq 0$, $u_i \geq 0$, and $\alpha_{ij} \geq 0$ are intrinsic to the agent. The damping d_{ij} represents how resistant agent i is to forming a preference for category j, the attention u_i represents how attentive agent i is to the preferences of other agents, and the self-reinforcement α_{ij} represents how resistant agent i is to changing its preference about category j.

The parameters a^a_{jk}, a^o_{jl}, and b_{ij} are extrinsic to the agent. The communication matrix $[a^a_{ik}]$ describes the communication strength between agent i and agent k, the belief matrix $[a^o_{jl}]$ describes the correlation of preferences for category j and category l, and the environmental input b_{ij} describes the impact of the environment on agent i's preference for category j. The saturating function $\mathcal{S}$ is selected such that $S(0) = 0$, $S'(0) = 1$, and $S'''(0) \neq 0$ (Leonard et al., 2024). We use $\tanh(\cdot)$ for the saturation function in our experiments.

For a given communication matrix $[a^o_{jl}]$ and belief matrix $[a^o_{jl}]$, the sensitivity of preference formation is determined by an agent's intrinsic parameters d_{ij}, u_i, and α_{ij}, and the equilibrium value of each preference is a function of the environmental input b_{ij} (see Figure 2).

Mutually exclusive categories. In the nonlinear opinion dynamics framework, the presence of mutually exclusive categories simplifies the model. For example, consider the case of $\mathcal{N}_a$ agents and $\mathcal{N}_o = 2$ categories. The categories are said to be mutually exclusive if $a^o_{12}, a^o_{21} \leq 0$ and $z_{i1} = -cz_{i2}$ (i.e. a positive preference for one category implies a negative preference for the other). In this setting, the dynamics for z_{i1} and z_{i2} decouple, and Equation (1) reduces to

$$\dot{z}_i = -d_i z_i + \mathcal{S}\left(u_i\left(\tilde{\alpha}_i z_i + \sum_{\substack{k=1\\k\neq i}}^{\mathcal{N}_\mathrm{a}} \tilde{a}_{ik} z_{kj}\right)\right) + b_i. \quad (2)$$

In our model, we leverage the presence of mutually exclusive categories to systematically identify opportunities for dimensionality reduction. Additional details are provided in Appendix D.

4 Method

In this section we present our BINN model for relational inference (see Figure 3). Given trajectories of a multi-agent system, our goal is to predict agent behavior in an interpretable way by inferring the intrinsic and extrinsic characteristics of agents as described by the nonlinear opinion dynamics model (see Section 3 and Appendix D).

We define a multi-agent system as a system of $\mathcal{N}_\mathrm{a} \in \mathbb{N}$ agents, each with real-valued preferences for a set of $\mathcal{N}_\mathrm{o} \in \mathbb{N}$ categories. We learn a mapping between the observed behavior of agents and their preferences for a set of latent categories using a graph neural network, and the parameters of a nonlinear opinion dynamics model which determine the evolution of agent preferences.

We model the evolution of agent preference by the dynamical equation presented in Equation (1). With this formulation, we can control agent preferences by varying the environmental input as shown in Figure 5.

We train our model using N trajectories of T observations, where each observation has dimension d. For a given trajectory, we denote the observed state of agent i at time t by the concatenation of position and velocity $\mathbf{x}_{i,t} = [\mathbf{p}_{i,t}, \mathbf{v}_{i,t}] \in \mathbb{R}^d$, and the preferences of agent i at time t by $\mathbf{z}_{i,t} \in \mathbb{R}^{\mathcal{N}_o}$.

4.1 Encoder

We use separate encoding networks to learn the mappings from agent states to agent preferences and from agent states to environmental inputs. We refer to the first of these encoders as the preference encoder E_z, and the latter as the environmental input encoder E_b. Each encoder takes state observations of the multi-agent system at time t as input and processes them in an message passing neural network (MPNN) (Gilmer et al., 2017). The multi-agent system is represented as a fully-connected graph with node values determined by the physical state of each agent.

Our preference encoder E_z performs the following message passing functions for agent i at timestep t:

$$\mathbf{z}'_{i,t} = f^z_{\mathrm{emb}}\left(\mathbf{x}_{i,t}\right), \quad (3)$$

$$v \to e: \quad \mathbf{m}^z_{(i,k),t} = f^z_{v\to e}\left(\mathbf{z}'_{i,t}, \mathbf{z}'_{k,t}\right), \quad (4)$$

$$e \to v: \quad \mathbf{z}_{i,t} = f^z_{e\to v}\left(\textstyle\sum_{k\neq i} \mathbf{m}^z_{(i,k),t}, \mathbf{z}'_{i,t}\right), \quad (5)$$

where f^z_{emb}, $f^z_{v\to e}$, and $f^z_{e\to v}$ are 3-layer MLPs. The environmental input encoder E_b is designed similarly, and performs the following message passing functions for agent i at timestep t:

$$\mathbf{b}'_{i,t} = f^b_{\mathrm{emb}}\left(\mathbf{x}_{i,t}\right), \quad (6)$$

$$v \to e: \quad \mathbf{m}^b_{(i,k),t} = f^b_{v\to e}\left(\mathbf{b}'_{i,t}, \mathbf{b}'_{k,t}\right), \quad (7)$$

$$e \to v: \quad \mathbf{b}_{i,t} = f^b_{e\to v}\left(\textstyle\sum_{k\neq i} \mathbf{m}^b_{(i,k),t}, \mathbf{b}'_{i,t}\right), \quad (8)$$

where f^b_{emb}, $f^b_{v\to e}$, and $f^b_{e\to v}$ are 3-layer MLPs.

4.2 Latent nonlinear opinion dynamics

We use the nonlinear opinion dynamics formulation in Equation (1) to model the evolution of agent preferences on a set of latent categories,

$$\dot{\mathbf{z}}_{i,t+1} = f_{\mathrm{NOD}}\left(\mathbf{z}_{i,t}, \mathbf{b}_t, \mathbf{A}^{\mathrm{a}}_t\right). \quad (9)$$

We learn the intrinsic agent parameters, $\mathbf{d}$, $\mathbf{u}$, and $\boldsymbol{\alpha}$, the extrinsic belief matrix $\mathbf{A}^{\mathrm{o}}$, and compute the inter-agent communication matrix $\mathbf{A}^{\mathrm{a}}_t$ at timestep t.

We compute future preferences using Euler integration,

$$\mathbf{z}_{i,t+1} = \mathbf{z}_{i,t} + f_{\mathrm{NOD}}\left(\mathbf{z}_{i,t}, \mathbf{b}_t, \mathbf{A}^{\mathrm{a}}_t\right)\Delta t, \quad (10)$$

where the timestep Δt is dataset dependent.

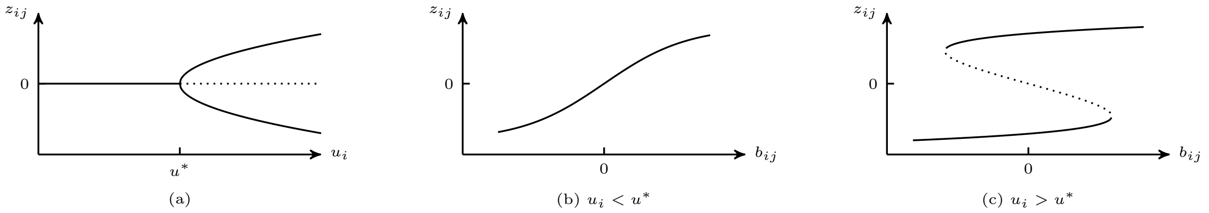


Figure 2: **Sensitivity to environmental inputs b_{ij}.** Solid lines represent stable equilibria and dotted line represents unstable equilibria. **(a)** We show the pitchfork bifurcation characteristic to Equation (1). The number, location, and stability of equilibria changes with the attention parameter u_i. **(b)** For attention $u_i < u^*$ preferences change linearly with environmental inputs. **(c)** For attention $u_i > u^*$ preferences change rapidly in response to environmental inputs with hysteresis encoding memory of previous states.

4.3 Communication matrix

We define the communication matrix $\mathbf{A}_t^{\mathrm{a}}$ as a function of inter-agent proximity. In our baseline BINN model, we define $\mathbf{A}_t^{\mathrm{a}}$ as the distance between observed positions,

$$a^{\mathrm{a}}_{(i,j),t} = \|\mathbf{p}_{i,t} - \mathbf{p}_{j,t}\|^2, \tag{11}$$

We also introduce *Behavior-Inspired Neural Network+* (BINN+), where we incorporate an informed hypothesis of how agents in the system communicate. For example, for systems in which agents have more influence on each other when they are closer (e.g., human interaction) we define $\mathbf{A}_t^{\mathrm{a}}$ as the inverse distance between observed positions,

$$a^{\mathrm{a}}_{(i,j),t} = \frac{1}{\|\mathbf{p}_{i,t} - \mathbf{p}_{j,t}\|^2 + \epsilon}, \tag{12}$$

where ϵ is a small constant.

4.4 Decoder

Our decoding network D_x, is an MPNN that maps agent preferences to predictions of agent states. At every timestep t, the latent preferences of the multi-agent system are represented as a fully-connected graph with node values determined by the preferences of each agent.

Our decoder D_x performs the following message passing functions for agent i at timestep t:

$$\hat{\mathbf{x}}'_{i,t} = f^x_{\mathrm{dec}}\left(\mathbf{z}_{i,t}\right), \tag{13}$$

$$v \to e: \quad \mathbf{m}^x_{(i,k),t} = f^x_{v\to e}\left(\hat{\mathbf{x}}'_{i,t}, \hat{\mathbf{x}}'_{k,t}\right), \tag{14}$$

$$e \to v: \quad \hat{\mathbf{x}}_{i,t} = f^x_{e\to v}\left(\textstyle\sum_{k\neq i} \mathbf{m}^x_{(i,k),t}, \hat{\mathbf{x}}'_{i,t}\right), \tag{15}$$

where f^x_{dec}, $f^x_{v\to e}$, and $f^x_{e\to v}$ are 3-layer MLPs.

4.5 Loss function

We train our model using the three component loss function,

$$\mathcal{L} = \mathcal{L}_{\mathrm{pred}} + \gamma_1 \mathcal{L}_{\mathrm{recon}} + \gamma_2 \mathcal{L}_{\mathrm{latent}} \tag{16}$$

where $\mathcal{L}_{\mathrm{pred}}$ is the prediction loss, $\mathcal{L}_{\mathrm{recon}}$ is the reconstruction loss and $\mathcal{L}_{\mathrm{latent}}$ is the latent loss.

The prediction loss, $\mathcal{L}_{\mathrm{pred}}$, is defined as the dissimilarity between the ground truth future state $\mathbf{x}_{i,t}$, and the predicted future state $\hat{\mathbf{x}}_{i,t}$,

$$\mathcal{L}_{\mathrm{pred}} = \frac{1}{T\mathcal{N}_a} \sum_{t=1}^{T} \sum_{i=1}^{\mathcal{N}_a} \left\| \mathbf{x}_{i,t} - \hat{\mathbf{x}}_{i,t} \right\|^2, \tag{17}$$

and encourages accuracy of future state prediction. The reconstruction loss, $\mathcal{L}_{\mathrm{recon}}$, is defined as the dissimilarity between the ground truth initial state $\mathbf{x}_{i,0}$, and the

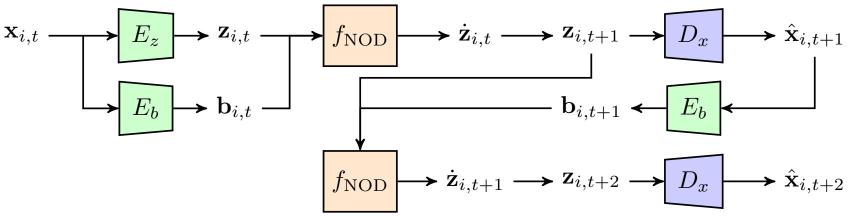


Figure 3: **Behavior-inspired neural network architecture overview.** Our network takes agent states as inputs and outputs predicted next states, while maintaining a representation of agent preferences for a set of latent categories. The network E_z encodes physical states to preferences for a set of latent categories and E_b encodes physical states into latent environmental inputs. In the latent space, we compute future preferences using the nonlinear opinion dynamics block f_{NOD}, and use the decoder D_x to map predicted preferences to predicted physical states. The latent dynamics are unrolled for multi-step trajectory prediction.

reconstructed initial state,

$$\mathcal{L}_{\text{recon}} = \frac{1}{\mathcal{N}_a} \sum_{i=1}^{\mathcal{N}_a} \left\| \mathbf{x}_{i,0} - \left(D_x \circ E_z\right)\left(\mathbf{x}_{i,0}\right) \right\|^2, \tag{18}$$

and encourages the decoder to function as the inverse of the encoder. The latent loss, $\mathcal{L}_{\text{latent}}$, is defined as the dissimilarity of preferences predicted by the dynamical model and those encoded by the preference encoder,

$$\mathcal{L}_{\text{latent}} = \frac{1}{T\mathcal{N}_a} \sum_{t=1}^{T} \sum_{i=1}^{\mathcal{N}_a} \left\| \Delta \mathbf{z}_{i,t+1} - f_{\text{NOD}}\left(\mathbf{z}_{i,t}\right) \right\|^2, \tag{19}$$

where $\Delta \mathbf{z}_{i,t+1} = \left(\mathbf{z}_{i,t+1} - \mathbf{z}_{i,t}\right) / \Delta t$. This loss encourages alignment between the representations encoded by E_z and those predicted by f_{NOD}.

5 Experiments

In this section, we highlight the utility of our model for discovering interpretable representations useful for dimensionality reduction, and its efficacy for long-horizon trajectory prediction. Dataset and training details are provided in Appendices A and B.

5.1 Interpretability of the latent space

We demonstrate the interpretability of our latent representations on both the pendulum and double pendulum datasets.

Pendulum. We model the pendulum as a single agent system represented as a graph with a single node. We construct a graphical representation at every timestep and set the node feature to the concatenation of the position and velocity of the pendulum bob.

The relationship between observed physical states and learned preferences is shown in Figure 4. Our model suggests the pendulum is a 1-dimensional system which is consistent with physical understanding. We can see this in two ways. First, since the preferences z_1 and z_2 switch signs when the pendulum bob changes direction, we can interpret the learned categories as a desire to move clockwise ($z_1 > 0$) and a desire to move counter-clockwise ($z_2 > 0$). Secondly, since the two preferences are perfectly out-of-phase and the entries of the learned belief matrix are negative,

$$\mathbf{A}^{\text{o}} = \begin{bmatrix} 0 & -1.2947 \\ -0.9147 & 0 \end{bmatrix}, \tag{20}$$

the conditions for mutual exclusivity is satisfied (see Section 3.1).

We empirically validate that the dynamics of this system can be captured with a 1-dimensional latent space. The prediction error for models with a 2-dimensional and 1-dimensional latent space are comparable i.e., 1.9718e−3 MSE and 1.9916e−3 MSE respectively. Qualitative results are shown in Figure 4 (bottom).

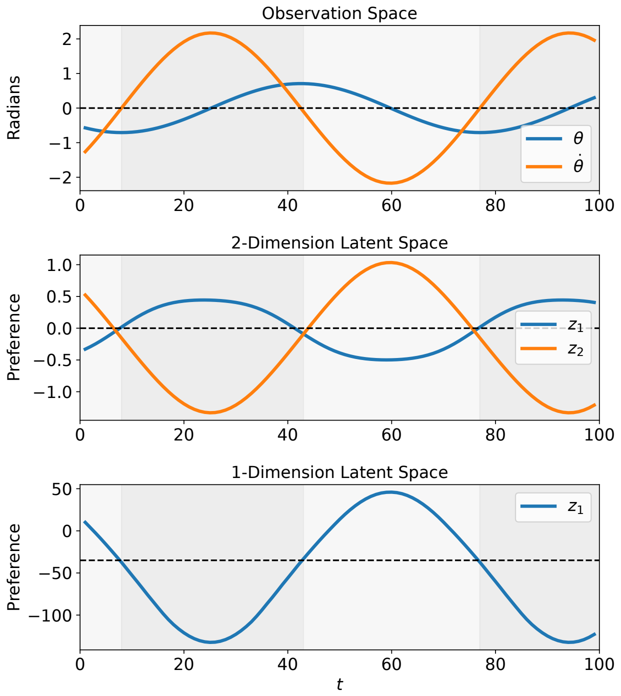
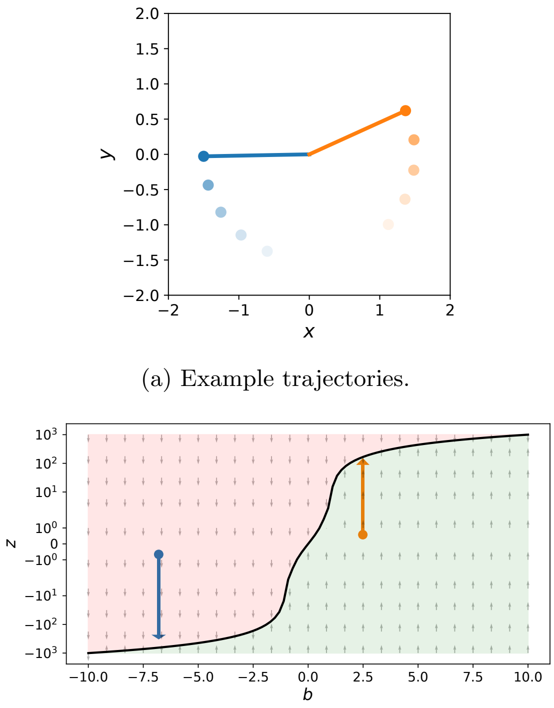


Figure 4: **Mutually exclusive pendulum preferences.** We show the observed, and latent representations of the pendulum bob. **(Top-bottom)** The observed position and velocity of the pendulum bob; the learned representation of agent preferences on a 2-dimensional latent space, and the learned representation of agent preferences on a 1-dimensional latent space. In the 2-dimensional space, preferences are out of phase indicating that they are mutually exclusive.

In addition to providing a mechanism for identifying opportunities for dimensionality reduction, the nonlinear opinion dynamics inductive bias also provides a mechanism for controlling preference formation through the environmental input b_{ij}. We show trajectories with different rotational directions in Figure 5a, and their corresponding environmental input latent preference pairs, (b_{ij}, z_{ij}), on the bifurcation diagram in Figure 5b. The initial preferences for both trajectories are near neutral, but the initial environmental inputs are far from zero. The negative environmental input drives the initial preference of the blue trajectory toward an equilibrium in the preference space that corresponds to a strongly negative preference, and a clockwise motion of the pendulum bob in the physical space. The positive environmental input drives the initial preference of the orange trajectory toward an equilibrium in the



(a) Example trajectories.

(b) Bifurcation diagram.

Figure 5: **Pendulum preference bifurcation diagram.** **(a)** We show a pendulum trajectory with clockwise motion (blue) and counterclockwise motion (orange). **(b)** The initial preferences are near neutral, but large environmental inputs drive preferences in opposite directions resulting in opposite motions in the physical space.

preference space that corresponds to a strongly positive preference, and a counterclockwise motion of the pendulum bob in the physical space.

Double pendulum. We use the double pendulum dataset to demonstrate interpretability in a more complex setting. We model the double pendulum as a two agent system represented as a graph with a two nodes. We construct a graphical representation at every timestep and set node features to the concatenation of the position and velocity of the corresponding double pendulum bob.

The relationship between the observed physical states and the learned preferences is shown in Figure 6. For the first bob, when the preference $z_{1,1}$ switches from greater (less) than to less (greater) than preference $z_{1,2}$, the physical bob switches from clockwise (counterclockwise) to counterclockwise (clockwise) motion. The opposite is true for the second bob. This behavior is observed across examples (see Appendix C.1). In contrast to pendulum preferences, double pendulum preferences are non-mutually exclusive. This observation is supported by the belief matrix, which does not satisfy the mutually-exclusive condition described in Section 3.1,

$$\mathbf{A}^{\mathrm{o}} = \begin{bmatrix} 0 & -2.8137 \\ 1.1597 & 0 \end{bmatrix}. \tag{21}$$

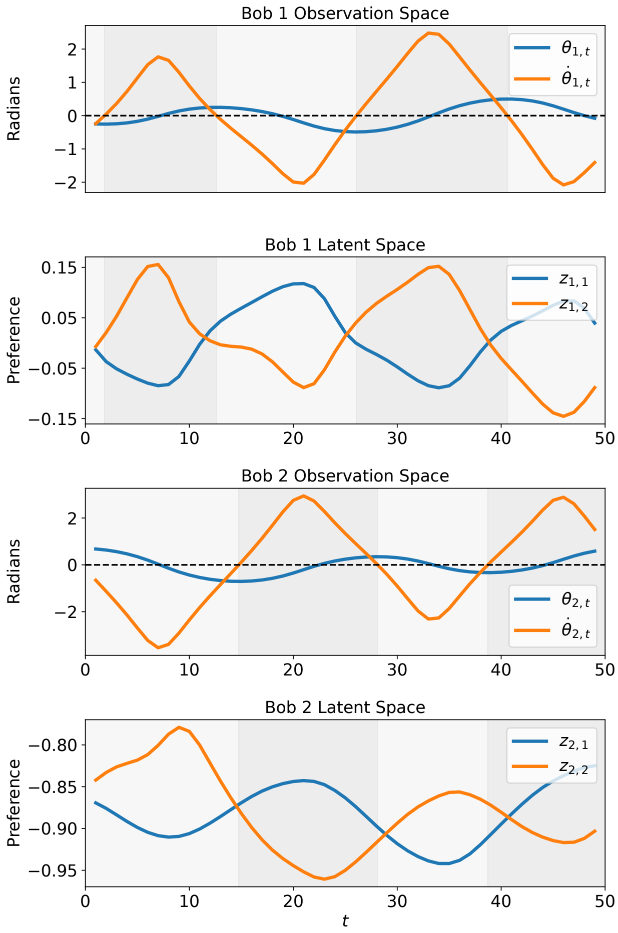


Figure 6: **Non-mutually exclusive double pendulum preferences.** We show the observed, and latent representations of each double pendulum bob. **(Top-bottom)** The observed position and velocity of the first bob; the learned preferences of the first bob; the observed position and velocity of the second bob; the learned preferences of the second bob. When the dominance of the preference switch, the physical bob's motion switches direction.

5.2 Trajectory prediction

We demonstrate the efficacy of our model for trajectory prediction on several datasets. For each system, we construct a fully-connected graph with $\mathcal{N}_a$ nodes, and initialize node features to the concatenation of the position and velocity of the corresponding agent. We report the quantitative performance of our model against competitive baselines in Table 1, and show the qualitative performance of our model in Figure 7.

Table 1: **Trajectory prediction.** Trajectory prediction error on the Mass-Spring, Kuramoto, TrajNet++, and NBA Player datasets is reported. Our BINN and BINN+ models outperform baseline models on all datasets.

Network	Mass-Spring	Kuramoto	TrajNet++	NBA Player
BINN*	$\mathbf{2.88{\times}10^{-4}}$	$\mathbf{4.98{\times}10^{-3}}$	2.16×10^{-2}	5.12×10^{-3}
BINN+*	$\mathbf{2.88{\times}10^{-4}}$	$\mathbf{4.98{\times}10^{-3}}$	$\mathbf{1.20{\times}10^{-2}}$	$\mathbf{4.19{\times}10^{-3}}$
NRI	3.73×10^{-3}	8.20×10^{-3}	8.49×10^{-2}	-
dNRI	4.86×10^{-3}	6.41×10^{-3}	4.00×10^{-2}	6.42×10^{-3}

Dataset details are provided in Appendix A.

5.2.1 Mechanical systems

Mass-spring. We demonstrate the utility of our model for trajectory prediction on the mass-spring dataset. The system consists of five masses and the dynamics are governed by a second-order linear equation.

We compute the communication matrices for our BINN and BINN+ models using the inter-agent distance defined in Equation (11). The predictive performance of our BINN models exceeds the performance of the NRI (Kipf et al., 2018) and dNRI (Graber & Schwing, 2020) models by an order of magnitude.

Kuramoto oscillator. The Kuramoto oscillator (Kuramoto, 1984) dataset describes a 5-agent system with dynamics governed by a first-order nonlinear equation.

We compute the communication matrices for our BINN and BINN+ models using the inter-agent distance defined in Equation (11). On this dataset, our BINN models perform comparably with baseline models.

5.2.2 Human behavior

TrajNet++ dataset. The TrajNet++ trajectory forecasting dataset (Kothari et al., 2021) is a collection of interaction-centric multi-agent datasets. We evaluate our model on a synthetic subset of the TrajNet++ dataset. This subset is comprised of 43,697 agent-agent centric trajectories. Each trajectory is 19-timesteps long and consists of five agents. Since Δt is not provided for the synthetic dataset, but is required by our model, we arbitrarily set $\Delta t = 1$ for all experiments.

We compute the communication matrix for our BINN model using the inter-agent distance defined in Equation (11). For our BINN+ model, we assume pedestrian sensing scales inversely with inter-agent distance and introduce a prior $\tilde{\mathbf{A}}^{a}_{(i,j),t}$ on the communication matrix,

$$\tilde{\mathbf{A}}^{a}_{(i,j),t} = \frac{1}{\|\mathbf{p}_{i,t} - \mathbf{p}_{j,t}\|^2 + \epsilon}. \tag{22}$$

We augment $\tilde{\mathbf{A}}^{a}_{(i,j),t}$ with a learned multiplier $\mathbf{A}^{\text{pre}}_{(i,j)}$, and form the communication matrix by way of their product,

$$\mathbf{A}^{a}_{(i,j),t} = \mathbf{A}^{\text{pre}}_{(i,j)} \odot \tilde{\mathbf{A}}^{a}_{(i,j),t}. \tag{23}$$

On this dataset, our BINN and BINN+ models outperform all baseline models. The relationship between the observed trajectories and learned preferences is shown in Figure 8. Three consistent trends emerge across examples. First, when the relative magnitudes of z_{i1} and z_{i2} change, the direction of travel changes from right to left. Second, when the relative magnitude of z_{i3} to z_{i4} increases, agents increase their y-velocity.

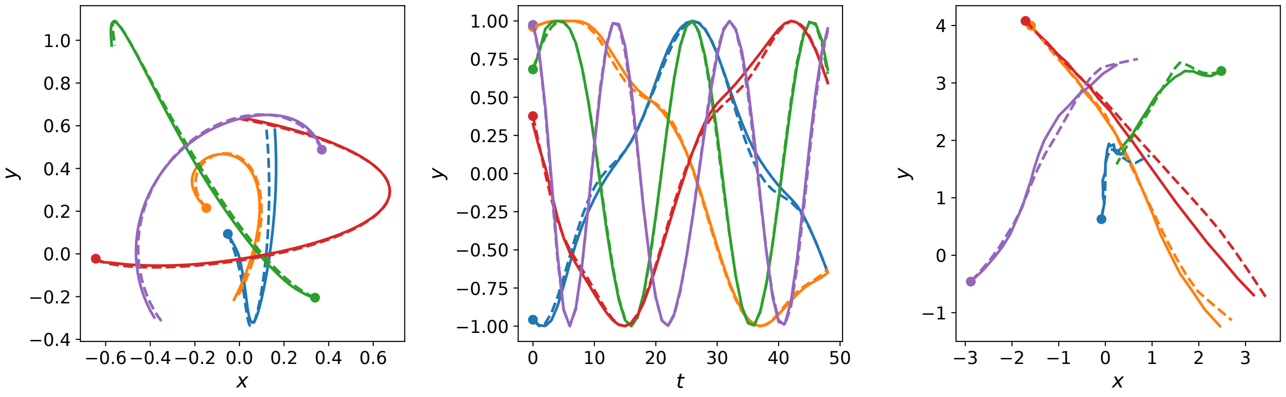


Figure 7: **Trajectory prediction.** We show trajectories predicted by BINN (solid) and the corresponding ground truths (dashed). **(Left-right)** Examples from mass-spring, Kuramoto oscillator, and TrajNet++ datasets.

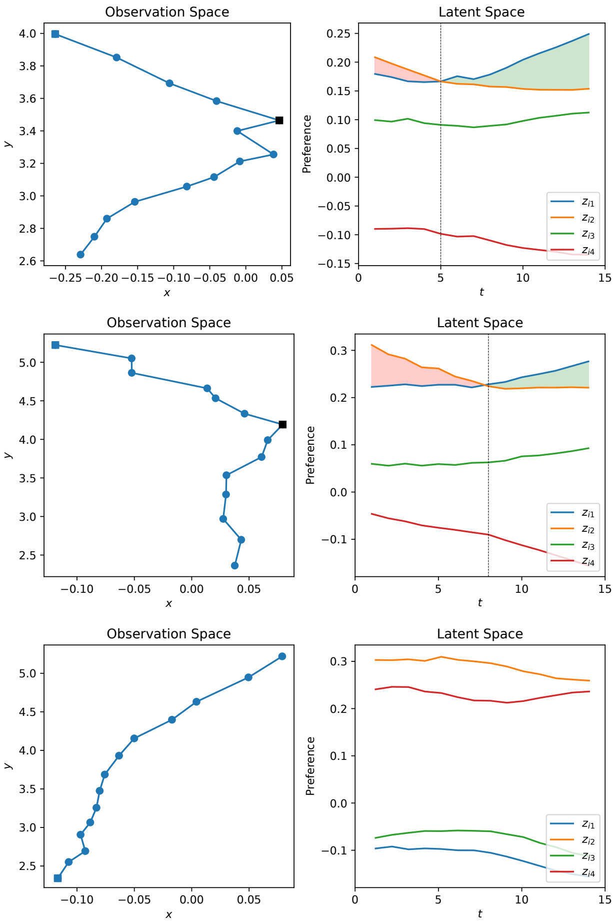


Figure 8: **TrajNet++ preferences. (Left)** We visualize example trajectories from the dataset for a single agent. The initial position of the trajectory is marked by a blue square and the point of direction change is marked by a black square. **(Right)** We plot the agent preferences for each latent categories. Dominant z_{i1} corresponds to leftwards motion and dominant z_{i2} corresponds to rightward motion. Dominant z_{i3} corresponds to downwards motion and dominant z_{i4} corresponds to upwards motion.

Finally, the dominant z_{i3} in row 1 and 2 results in downwards motion, while the dominant z_{i4} in row 3 results in upwards motion.

NBA player dataset. The SportUV dataset (Linou et al., 2016) is comprised of 358,217 real-world NBA player trajectories. Each trajectory is 15 timestep long and consists of 10 agents.

We compute the communication matrix for our BINN model using the inter-agent distance defined in Equations (11), and the communication matrix for our BINN+ model using Equation (22) and (23). Our BINN+ model performs comparably to the dNRI model. The relationship between the observed trajectories and learned preferences is shown in Appendix C.2, Figure 10.

Table 2: **Robustness to choice of latent dimension.** We compare the trajectory prediction performance of our BINN model with different latent dimensions. We use a lower latent dimension in BINN-small and a higher latent dimension in BINN-large. The predictive performance across models is comparable.

Dimension	Mass-Spring	Kuramoto	TrajNet++
BINN-small	$\mathbf{2.88{\times}10^{-4}}$	4.98×10^{-3}	$\mathbf{1.20{\times}10^{-2}}$
BINN-large	3.33×10^{-4}	$\mathbf{4.52{\times}10^{-3}}$	2.21×10^{-2}

5.3 Robustness to choice of latent dimension

We investigate the robustness of our BINN model to the choice of latent dimension. We report the performance of our BINN model for large and small latent dimension in Table 2 . Across experiments, the latent dimension of BINN-large is twice that of BINN-small. The performance variation across models is minimal for all experiments, demonstrating the robustness of our model to the choice of latent dimension. We attribute this robustness to the nonlinear opinion dynamics inductive bias which imparts a mechanism for identifying mutually exclusive categories. We report the latent dimension used in each experiment in Appendix B, Table 5.

6 Conclusion

In this paper, we address the challenge of designing a relational inference model with inter-agent relationships that are informed by preferences for multiple categories. To do this, we propose the use of nonlinear opinion dynamics as an inductive bias, and introduce an interpretable and flexible relational inference model. In addition to outperforming competitive existing relational inference models on multiple trajectory prediction datasets, our model provides a mechanisms for identification of mutually exclusive categories, and controlling preference formation.

Limitations and future work. Similarly to prior work, our BINN models assume the same agents are present throughout a trajectory. Efforts to address this limitation would broaden the applicability of this work.

Acknowledgements

This work was partially supported by NSF grant MRSEC DMR-2011750 and the Princeton Catalysis Initiative. We sincerely thank the AISTATS 2025 reviewers and area chair for their thoughtful feedback.

References

Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 961–971, 2016.

Ferran Alet, Erica Weng, Tomás Lozano-Pérez, and Leslie P Kaelbling. Neural relational inference with fast modular meta-learning. *Advances in Neural Information Processing Systems*, 32, 2019.

Christine Allen-Blanchette. Hamiltonian gan. In *6th Annual Learning for Dynamics & Control Conference*, pp. 1662–1674. PMLR, 2024.

Claudio Altafini. Consensus problems on networks with antagonistic interactions. *IEEE transactions on automatic control*, 58(4):935–946, 2012.

Michele Ballerini, Nicola Cabibbo, Raphael Candelier, Andrea Cavagna, Evaristo Cisbani, Irene Giardina, Vivien Lecomte, Alberto Orlandi, Giorgio Parisi, Andrea Procaccini, et al. Interaction ruling animal collective behavior depends on topological rather than metric distance: Evidence from a field study. *Proceedings of the national academy of sciences*, 105 (4):1232–1237, 2008.

Anastasia Bizyaeva, Giovanna Amorim, María Santos, Alessio Franci, and Naomi Ehrich Leonard. Switching transformations for decentralized control of opinion patterns in signed networks: Application to dynamic task allocation. *IEEE Control Systems Letters*, 6: 3463–3468, 2022a.

Anastasia Bizyaeva, Alessio Franci, and Naomi Ehrich Leonard. Nonlinear opinion dynamics with tunable sensitivity. *IEEE Transactions on Automatic Control*, 68(3):1415–1430, 2022b.

Anastasia Bizyaeva, Marcela Ordorica Arango, Yunxiu Zhou, Simon Levin, and Naomi Ehrich Leonard. Active risk aversion in sis epidemics on networks. In *2024 American Control Conference (ACC)*, pp. 4428–4433. IEEE, 2024.

Antoine Browaeys and Thierry Lahaye. Many-body physics with individually controlled rydberg atoms. *Nature Physics*, 16(2):132–142, 2020.

Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the national academy of sciences*, 113 (15):3932–3937, 2016.

Francesco Bullo. *Lectures on network systems*, volume 1. CreateSpace, 2018.

Sergio Casas, Wenjie Luo, and Raquel Urtasun. Intentnet: Learning to predict intention from raw sensor data. In *Conference on Robot Learning*, pp. 947–956. PMLR, 2018.

Charlotte Cathcart, María Santos, Shinkyu Park, and Naomi Ehrich Leonard. Opinion-driven robot navigation: Human-robot corridor passing. 2022.

Siyuan Chen, Jiahai Wang, and Guoqing Li. Neural relational inference with efficient message passing mechanisms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 7055–7063, 2021.

Pranav Dandekar, Ashish Goel, and David T Lee. Biased assimilation, homophily, and the dynamics of polarization. *Proceedings of the National Academy of Sciences*, 110(15):5791–5796, 2013.

Abhimanyu Das, Sreenivas Gollapudi, and Kamesh Munagala. Modeling opinion dynamics in social networks. In *Proceedings of the 7th ACM international conference on Web search and data mining*, pp. 403–412, 2014.

Nachiket Deo and Mohan M Trivedi. Convolutional social pooling for vehicle trajectory prediction. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 1468–1476, 2018.

Edward Fiorelli, Naomi Ehrich Leonard, Pradeep Bhatta, Derek A Paley, Ralf Bachmayer, and David M Fratantoni. Multi-auv control and adaptive sampling in monterey bay. *IEEE journal of oceanic engineering*, 31(4):935–948, 2006.

Jakob N Foerster, Yannis M Assael, Nando de Freitas, and Shimon Whiteson. Learning to communicate to solve riddles with deep distributed recurrent q-networks. *arXiv preprint arXiv:1602.02672*, 2016.

Alessio Franci, Anastasia Bizyaeva, Shinkyu Park, and Naomi Ehrich Leonard. Analysis and control of agreement and disagreement opinion cascades. *Swarm Intelligence*, 15(1):47–82, 2021.

Nicholas Galioto and Alex Arkady Gorodetsky. Bayesian system id: optimal management of parameter, model, and measurement uncertainty. *Nonlinear Dynamics*, 102(1):241–267, 2020.

Jiyang Gao, Chen Sun, Hang Zhao, Yi Shen, Dragomir Anguelov, Congcong Li, and Cordelia Schmid. Vectornet: Encoding hd maps and agent dynamics from vectorized representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11525–11533, 2020.

Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pp. 1263–1272. PMLR, 2017.

Francesco Giuliari, Irtiza Hasan, Marco Cristani, and Fabio Galasso. Transformer networks for trajectory forecasting. In *2020 25th international conference on pattern recognition (ICPR)*. IEEE, 2021.

Martin Golubitsky, Ian Stewart, and David G Schaeffer. *Singularities and Groups in Bifurcation Theory: Volume II*, volume 69. Springer Science & Business Media, 2012.

Colin Graber and Alexander G Schwing. Dynamic neural relational inference. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8513–8522, 2020.

Samuel Greydanus, Misko Dzamba, and Jason Yosinski. Hamiltonian neural networks. *Advances in neural information processing systems*, 32, 2019.

John Guckenheimer and Philip Holmes. *Nonlinear oscillations, dynamical systems, and bifurcations of vector fields*, volume 42. Springer Science & Business Media, 2013.

Emanuel Gull, Olivier Parcollet, and Andrew J Millis. Superconductivity and the pseudogap in the two-dimensional hubbard model. *Physical review letters*, 110(21):216405, 2013.

Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018.

Sandro Hauri, Nemanja Djuric, Vladan Radosavljevic, and Slobodan Vucetic. Multi-modal trajectory prediction of nba players. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1640–1649, 2021.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 1997.

Hanna Zsofia Horvath and Denes Takacs. Stability and local bifurcation analyses of two-wheeled trailers considering the nonlinear coupling between lateral and vertical motions. *Nonlinear Dynamics*, 2022.

Haimin Hu, Kensuke Nakamura, Kai-Chieh Hsu, Naomi Ehrich Leonard, and Jaime Fernández Fisac. Emergent coordination through game-induced nonlinear opinion dynamics. In *2023 62nd IEEE Conference on Decision and Control (CDC)*, pp. 8122–8129. IEEE, 2023.

Eric W Justh and PS Krishnaprasad. Natural frames and interacting particles in three dimensions. In *Proceedings of the 44th IEEE Conference on Decision and Control*, pp. 2841–2846. IEEE, 2005.

Diederik P Kingma, Max Welling, et al. Auto-encoding variational bayes, 2013.

Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. Neural relational inference for interacting systems. In *International conference on machine learning*, pp. 2688–2697. Pmlr, 2018.

Parth Kothari, Sven Kreiss, and Alexandre Alahi. Human trajectory forecasting in crowds: A deep learning perspective. *IEEE Transactions on Intelligent Transportation Systems*, 23(7):7386–7400, 2021.

Yoshiki Kuramoto. *Chemical Turbulence*. Springer, 1984.

Pablo Lemos, Niall Jeffrey, Miles Cranmer, Shirley Ho, and Peter Battaglia. Rediscovering orbital mechanics with machine learning. *Machine Learning: Science and Technology*, 4(4):045002, 2023.

Naomi E Leonard, Derek A Paley, Russ E Davis, David M Fratantoni, Francois Lekien, and Fumin Zhang. Coordinated control of an underwater glider fleet in an adaptive ocean sampling field experiment in monterey bay. *Journal of Field Robotics*, 27(6): 718–740, 2010.

Naomi Ehrich Leonard, Derek A Paley, Francois Lekien, Rodolphe Sepulchre, David M Fratantoni, and Russ E Davis. Collective motion, sensor networks, and ocean sampling. *Proceedings of the IEEE*, 95(1):48–74, 2007.

Naomi Ehrich Leonard, Keena Lipsitz, Anastasia Bizyaeva, Alessio Franci, and Yphtach Lelkes. The nonlinear feedback dynamics of asymmetric political polarization. *Proceedings of the National Academy of Sciences*, 118(50):e2102149118, 2021.

Naomi Ehrich Leonard, Anastasia Bizyaeva, and Alessio Franci. Fast and flexible multiagent decision-making. *Annual Review of Control, Robotics, and Autonomous Systems*, 7, 2024.

Herbert Levine, Wouter-Jan Rappel, and Inon Cohen. Self-organization in systems of self-propelled particles. *Physical Review E*, 63(1):017101, 2000.

Kostya Linou, Dzmitryi Linou, and Martijn de Boer. Visualization of nba games from raw sportvu data logs. git@github.com:linouk23/NBA-Player-Movements.git, 2016.

Sindy Löwe, David Madras, Richard Zemel, and Max Welling. Amortized causal discovery: Learning to infer causal graphs from time-series data. In *Conference on Causal Learning and Reasoning*, pp. 509–525. PMLR, 2022.

Karttikeya Mangalam, Harshayu Girase, Shreyas Agarwal, Kuan-Hui Lee, Ehsan Adeli, Jitendra Malik, and Adrien Gaidon. It is not the journey but the destination: Endpoint conditioned trajectory prediction. In *Computer Vision–ECCV 2020: 16th European*

Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16, 2020.

Justice J Mason, Christine Allen-Blanchette, Nicholas Zolman, Elizabeth Davison, and Naomi Leonard. Learning interpretable dynamics from images of a freely rotating 3d rigid body. *arXiv preprint arXiv:2209.11355*, 2022.

Justice J Mason, Christine Allen-Blanchette, Nicholas Zolman, Elizabeth Davison, and Naomi Ehrich Leonard. Learning to predict 3d rotational dynamics from images of a rigid body with unknown mass distribution. *Aerospace*, 10(11):921, 2023.

Marcela Ordorica Arango, Anastasia Bizyaeva, Simon A Levin, and Naomi Ehrich Leonard. Opinion-driven risk perception and reaction in sis epidemics. *arXiv e-prints*, pp. arXiv–2410, 2024.

Juan A Paredes and Dennis S Bernstein. Identification of self-excited systems using discrete-time, time-delayed lur'e models. In *2021 American Control Conference (ACC)*, pp. 3939–3944. IEEE, 2021.

Juan A Paredes and Dennis S Bernstein. Experimental application of predictive cost adaptive control to thermoacoustic oscillations in a rijke tube. *arXiv preprint arXiv:2402.00346*, 2024.

Juan A Paredes, Yulong Yang, and Dennis S Bernstein. Output-only identification of self-excited systems using discrete-time lur'e models with application to a gas-turbine combustor. *International Journal of Control*, 97(2):187–212, 2024.

Mohammad Reza Rezaei and Adji Bousso Dieng. Alternators for sequence modeling. *arXiv preprint arXiv:2405.11848*, 2024.

Riley J Richards, Yulong Yang, Juan A Paredes, and Dennis S Bernstein. Output-only identification of lur'e systems with hysteretic feedback nonlinearities. In *2024 American Control Conference (ACC)*, pp. 2891–2896. IEEE, 2024.

Wilbert Samuel Rossi and Paolo Frasca. Opinion dynamics with topological gossiping: Asynchronous updates under limited attention. *IEEE Control Systems Letters*, 4(3):566–571, 2020.

Amir Sadeghian, Vineet Kosaraju, Ali Sadeghian, Noriaki Hirose, Hamid Rezatofighi, and Silvio Savarese. Sophie: An attentive gan for predicting paths compliant to social and physical constraints. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1349–1358, 2019.

Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. A simple neural network module for relational reasoning. *Advances in neural information processing systems*, 30, 2017.

Thomas D. Seeley, P. Kirk Visscher, Thomas Schlegel, Patrick M. Hogan, Nigel R. Franks, and James A. R. Marshall. Stop signals provide cross inhibition in collective decision-making by honeybee swarms. *Science*, 335(6064):108–111, 2012.

Steven H Strogatz. *Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering*. CRC press, 2018.

Sainbayar Sukhbaatar, Rob Fergus, et al. Learning multiagent communication with backpropagation. *Advances in neural information processing systems*, 29, 2016.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27, 2014.

Anirudh Vemula, Katharina Muelling, and Jean Oh. Social attention: Modeling attention in human crowds. In *2018 IEEE international Conference on Robotics and Automation (ICRA)*, pp. 4601–4607. IEEE, 2018.

Pablo Villanueva-Domingo, Francisco Villaescusa-Navarro, Daniel Anglés-Alcázar, Shy Genel, Federico Marinacci, David N Spergel, Lars Hernquist, Mark Vogelsberger, Romeel Dave, and Desika Narayanan. Inferring halo masses with graph neural networks. *The Astrophysical Journal*, 935(1):30, 2022.

Aoran Wang and Jun Pang. Structural inference with dynamics encoding and partial correlation coefficients. In *12th International Conference on Learning Representations (ICLR'24)*. OpenReview. net, 2024.

Aoran Wang, Tsz Pan Tong, and Jun Pang. Effective and efficient structural inference with reservoir computing. In *International Conference on Machine Learning*, pp. 36391–36410. PMLR, 2023.

Keqin Wang, Yulong Yang, Ishan Saha, and Christine Allen-Blanchette. Understanding oversmoothing in gnns as consensus in opinion dynamics. *arXiv preprint arXiv:2501.19089*, 2025.

Zhe Wang, Petar Veličković, Daniel Hennes, Nenad Tomašev, Laurel Prince, Michael Kaisers, Yoram Bachrach, Romuald Elie, Li Kevin Wenliang, Federico Piccinini, et al. Tacticai: an ai assistant for football tactics. *Nature communications*, 15(1), 2024.

Ezra Webb, Ben Day, Helena Andres-Terre, and Pietro Lió. Factorised neural relational inference for multi-interaction systems. *arXiv preprint arXiv:1905.08721*, 2019.

Chenxin Xu, Maosen Li, Zhenyang Ni, Ya Zhang, and Siheng Chen. Groupnet: Multiscale hypergraph neural networks for trajectory prediction with relational reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6498–6507, 2022.

Chenxin Xu, Robby T Tan, Yuhong Tan, Siheng Chen, Yu Guang Wang, Xinchao Wang, and Yanfeng Wang. Eqmotion: Equivariant multi-agent motion prediction with invariant interaction reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1410–1420, 2023.

George F Young, Luca Scardovi, Andrea Cavagna, Irene Giardina, and Naomi E Leonard. Starling flock networks manage uncertainty in consensus at low cost. *PLoS computational biology*, 9(1):e1002894, 2013.

Cunjun Yu, Xiao Ma, Jiawei Ren, Haiyu Zhao, and Shuai Yi. Spatio-temporal graph transformer networks for pedestrian trajectory prediction. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16*, pp. 507–523. Springer, 2020.

Ye Yuan, Xinshuo Weng, Yanglan Ou, and Kris M Kitani. Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9813–9823, 2021.

Shuhan Zheng, Ziqiang Li, Kantaro Fujiwara, and Gouhei Tanaka. Diffusion model for relational inference. *arXiv preprint arXiv:2401.16755*, 2024.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model: **Yes. Theoretical background for opinion dynamics is described in Section 3 and Appendix D and description of the algorithm is given in Section 4 and Figure 3.**
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm: **Yes. Memory requirement for the proposed method and comparison with baselines are given in Table 4.**
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. **Code is open sourced at `https://github.com/CAB-Lab-Princeton/Behavior-Inspired-Neural-Network`.**

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results: **Yes. Assumptions we make about the nonlinear opinion dynamics model are given in Section 3 and Appendix D.**
   (b) Complete proofs of all theoretical results. **Not applicable. Theoretical results are not part of the contribution to this paper.**
   (c) Clear explanations of any assumptions: **N/A.**

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL): **Yes. All data generation and training hyper-parameters are given in Table 3 and 5. Code is open sourced.**
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen): **Yes. Training details are given in Table 5.**
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times): **N/A.**
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider): **Yes. Information is provided in Appendix B. Specific information on the Neuronic cluster is availible at `https://clusters.cs.princeton.edu`.**

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets: **Yes. TrajNet++ (Kothari et al., 2021) and SportsVU (Linou et al., 2016) are cited.**
   (b) The license information of the assets, if applicable: **N/A.**
   (c) New assets either in the supplemental material or as a URL, if applicable: **N/A.**
   (d) Information about consent from data providers/curators: **N/A.**
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content: **N/A.**

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots: **N/A.**
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable: **N/A.**
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation: **N/A.**

Appendix

A Datasets

A.1 Mechanical systems

All simulated mechanical system datasets were generated similarly. For each system, we generate 50,000 training trajectories, and 12,500 validation and testing trajectories. We note the numerical integrator, timestep, trajectory length and system parameters used to simulate each system in Table 3. We coarsen each simulated trajectory to reduce dataset size, and training time; the coarsening frequency for each system is provided in Table 3.

Pendulum. We generate a synthetic dataset of pendulum motion, using the second-order dynamical equation,

$$\ddot{\theta} = -\frac{g}{\ell}\sin\theta, \tag{24}$$

with $\ell = 1.0$ and gravity $g = 9.81$. For each trajectory, we sample initial conditions from the uniform distribution on the interval $[-0.5\pi, 0.5\pi]$.

Double pendulum. We generate a synthetic dataset of double pendulum motion, using the second-order dynamical equation,

$$(m_1 + m_2)\, l_1\ddot{\theta}_1 + m_2 l_2 \ddot{\theta}_2 \cos(\theta_2 - \theta_1) = m_2 l_2 \dot{\theta}_2^2 \sin(\theta_2 - \theta_1) - (m_1 + m_2)\, g \sin\theta_1, \tag{25}$$

$$l_2\ddot{\theta}_2 + l_1\ddot{\theta}_1 \cos(\theta_2 - \theta_1) = -l_1\dot{\theta}_1^2 \sin(\theta_2 - \theta_1) - g\sin\theta_2, \tag{26}$$

with $\ell_1 = \ell_2 = 1.0$, $m_1 = m_2 = 1.0$, and gravity $g = 9.81$. For each trajectory, we sample initial conditions from the uniform distribution on the interval $[-0.5\pi, 0.5\pi]$.

Mass-spring. We generate a synthetic dataset of mass-spring motion, using the second-order linear dynamical equation,

$$\ddot{\mathbf{r}}_i = \sum_{i\neq j}^{\mathcal{N}_a} -k_{ij}\left(\mathbf{r}_i - \mathbf{r}_j\right), \tag{27}$$

where the spring constant k_{ij} is either 2.5 or 0, and $k_{ij} = k_{ji}$. The initial conditions are sampled from a normal distribution with $\mu = 0$ and $\sigma = 0.3$.

Kuramoto oscillator. We generate a synthetic dataset of Kuramoto oscillator (Kuramoto, 1984) motion where the oscillator dynamics are defined by the first-order nonlinear equation,

$$\dot{\phi}_i = \omega_i + \sum_{i\neq j} k_{ij} \sin\left(\phi_i - \phi_j\right), \tag{28}$$

where the coupling constant k_{ij} is either 2.5 or 0 and $k_{ij} = k_{ji}$. The initial conditions are sampled from a normal distribution with $\mu = 0$ and $\sigma = 2\pi$.

A.2 Human behavior

TrajNet++ dataset. The TrajNet++ dataset Kothari et al. (2021) is divided into real and synthetic pedestrian trajectory datasets; we use the synthetic dataset in our experiments. The synthetic dataset is comprised of $43,697$ trajectories, and each trajectory has 19 timesteps and five agents.

Table 3: **Data generation and post processing parameters.**

	Pendulum	Double Pendulum	Mass-Spring	Kuramoto	TrajNet++
Integrator	RK4	RK4	RK4	RK4	n/a
Δt	1e-3	5e-4	5e-4	5e-4	1.0
Steps	5000	5000	5000	500	n/a
Coarsening	100	100	100	10	n/a

Table 4: **Computational cost.** Memory usage for BINN (ours), NRI, and dNRI is reported. Our model (BINN) has the lowest memory usage for all datasets (lower is better).

Network	Mass-Spring	Kuramoto	TrajNet++
BINN*	**496** MiB	**484** MiB	**406** MiB
NRI	1670 MiB	1658 MiB	916 MiB
dNRI	5008 MiB	4996 MiB	2180 MiB

NBA player movement dataset. The SportsVU dataset Linou et al. (2016) consists of NBA player trajectories captured over 631 games in the 2015/16 season. The dataset consists of 358, 217 trajectories, and each trajectory has 15 timesteps and 11 agents (10 players and the ball). We train our model on player trajectories and omit the ball.

B Training Details

In this section, we provide training details for each experiment. Our source code is publicly available at `https://github.com/CAB-Lab-Princeton/Behavior-Inspired-Neural-Network`.

All models were trained on Dell Precision 7920 work stations. Each work station is equipped with an Intel Xeon Gold 5220R 24 core CPU, two Nvidia A6000 GPUs, and 256GB of RAM.

We trained each model on a single GPU with CPU and RAM shared between two workloads. We report memory usage in Table 4 and training hyperparameters in Table 5. Memory usage is measured at the end of the first training epoch using NVIDIA-smi.

C Additional Experimental Results

C.1 Consistency across random seeds

The character of the preferences learned in our BINN model is consistent across random seeds. In Figure 9, we provide supplemental observation space to preference encodings for the double pendulum system presented in Section 5. The encodings on the left were learned with a random seed of 172, and the encodings on the right were learned with a random seed of 272. The double pendulum encodings in Figure 6 were learned with a random seed of 72.

C.2 Capturing real-world human behavior

Human behavior can be less predictable than simulated systems. To assess how well our model can capture the behavior of human agents, we performed additional experiments on the SportsVU dataset Linou et al. (2016). We

Table 5: **Experiment hyperparameters.**

Hyper-parameter	Pendulum	Double Pendulum	Mass-Spring	Kuramoto	TrajNet++
$\mathcal{N}_\text{a}$	1	2	5	5	5
$\mathcal{N}_\text{o}$	2	2	4	2	4
Epoch	500	500	1000	1000	1000
Batch Size	256	256	256	256	256
Activation	ReLU	ELU	tanh	tanh	tanh
Optimizer	Adam	Adam	Adam	Adam	Adam
Initial Learning Rate	1e-3	1e-3	1e-3	1e-3	5e-4
Hidden Dimension	64	64	128	128	128
Scheduler	n/a	n/a	StepLR	StepLR	StepLR
Scheduler Step	n/a	n/a	200	200	200
Scheduler Gamma	n/a	n/a	0.25	0.25	0.25
γ_1	1.0	1.0	1.0	1.0	1.0
γ_2	1.0	1.0	1.0	1.0	1.0

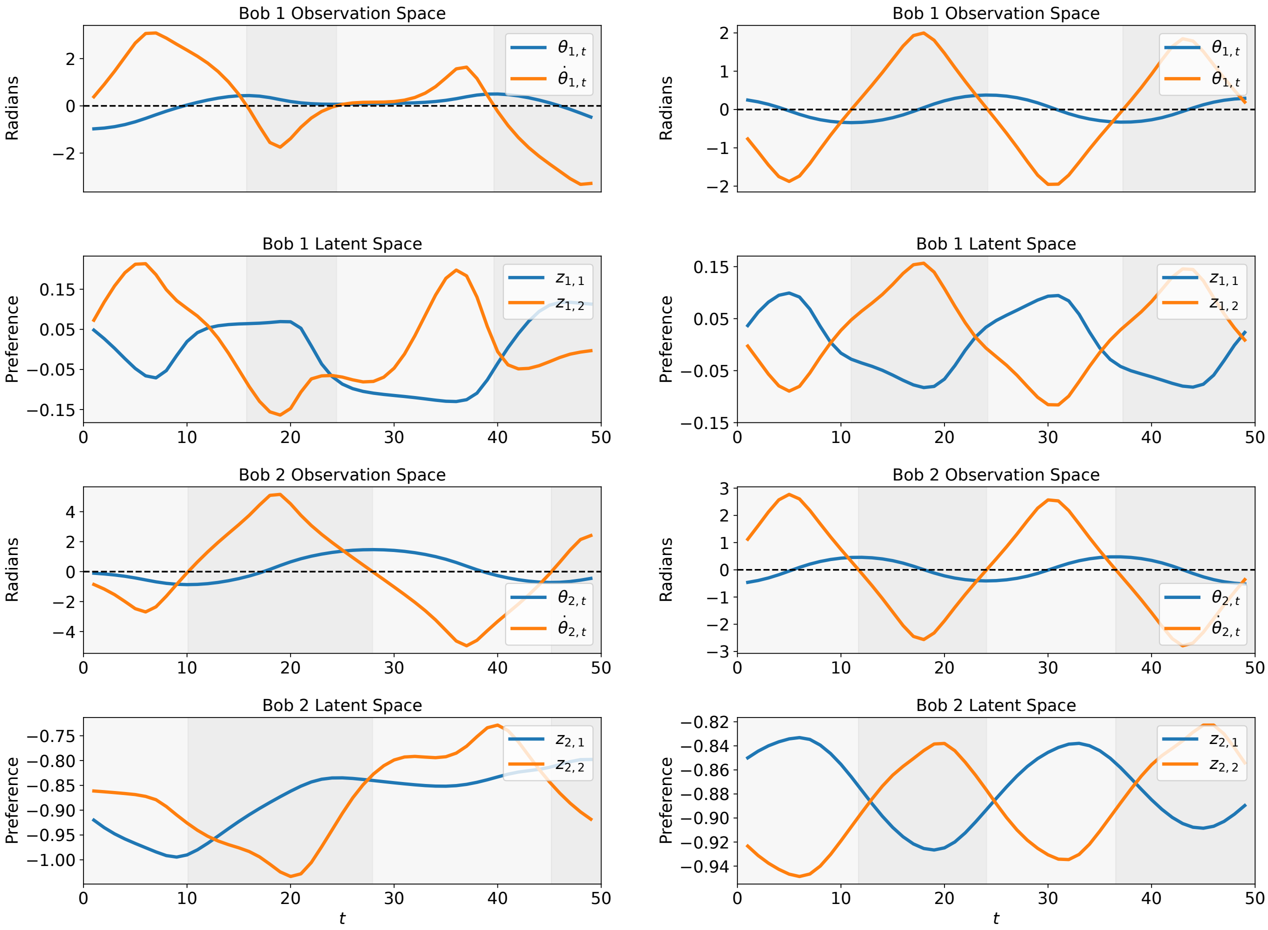


Figure 9: **Consistency across random seeds.** We show the observed, and latent representations of each double pendulum bob. **(Left-right)** Encodings learned with a random seed of 172; encodings learned with a random seed of 272. **(Top-bottom)** The observed position and velocity of the first bob; the learned preferences of the first bob; the observed position and velocity of the second bob; the learned preferences of the second bob. When preference switch, physical bob's motion switches direction.

provide observation space to preference encodings for the SportsVU dataset Linou et al. (2016) in Figure 10. The learned relationship between preferences is similar to the learned relationship for synthetic pedestrian data (see Section 5.2.2). When the relative magnitudes of z_{i1} and z_{i2} change, the direction of travel changes from upwards to downwards. When the relative magnitudes of z_{i3} and z_{i4} change, the direction of travel changes from leftwards to rightwards.

C.3 Additional baseline comparisons

We report an additional comparison against Factorised Neural Relational Inference (fNRI) (Webb et al., 2019). Similarly to our approach, fNRI models multiple types of inter-agent relationships; however, their approach uses a factorized graph representation, while ours uses the nonlinear opinion dynamics inductive bias. We report the 10-step trajectory prediction performance of our BINN+ and fNRI on the TrajNet++ dataset in Table 6, where our BINN+ model outperforms fNRI.

D Aspects of Bifurcation Theory and Nonlinear Opinion Dynamics

In this section, we aim provide intuition building examples for the pitchfork bifurcation introduced in Figure 2, and the notion of mutually exclusive categories introduced in Section 3.1.

Pitchfork bifurcation. Bifurcation theory can be described as the study of equations with multiple solutions.

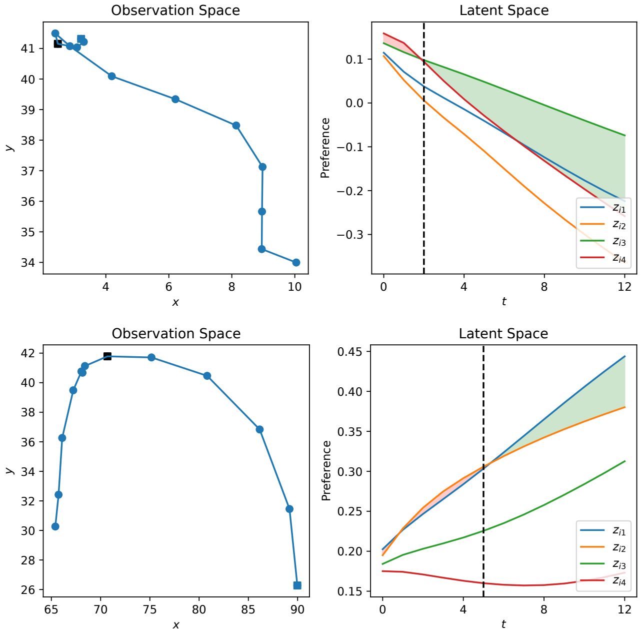


Figure 10: **NBA player preferences. (Left)** We visualize example trajectories from the dataset. The initial position of the trajectory is marked by a blue square and the point of direction change is marked by a black square. **(Right)** We plot agent preferences for each latent categories. Dominant z_{i1} corresponds to upward motion, and dominant z_{i2} corresponds to downward motion. Dominant z_{i3} corresponds to rightward motion, and dominant z_{i4} corresponds to leftward motion.

Consider, for example, the dynamical system defined by

$$\dot{z} = z^3 - uz. \tag{29}$$

The equilibrium solutions of this dynamical equation (i.e., the values of z for which $\dot{z} = 0$) are shown in the bifurcation diagram of Figure 11. When the bifurcation parameter u is less than or equal to the critical value $u^* = 0$, there is only one solution for Equation 29 (i.e., $z = 0$). When u is greater than u^*, the number of solutions jumps to three (i.e., $z \in \{\sqrt{u}, -\sqrt{u}, 0\}$). The jump in the number of solutions from one to three at the critical value u^*, is the basic property that characterizes a pitchfork bifurcation. As it turns out, Equation 29 is the simplest equation with this behavior (Golubitsky et al., 2012).

The stability of solutions to the dynamical system defined in Equation (29) can be determined by evaluating the derivative

$$\ddot{z} = 3z^2 - u. \tag{30}$$

When the derivative is greater than zero, the solution is stable. When it is less than zero, the solution is unstable. For the solution, $z = 0$, the derivative evaluates to

$$\ddot{z} = -u, \tag{31}$$

which is stable for $u < 0$, and unstable for $u > 0$. The change in the stability of the solution $z = 0$ is illustrated in the bifurcation diagram of Figure 11. When $u < 0$, the solution $z = 0$ is illustrated with a solid line and when

Table 6: **Trajectory prediction.** Trajectory prediction error on the TrajNet++ datasets is reported. Our BINN+ model outperforms fNRI.

Network	TrajNet++
BINN+*	$\mathbf{3.51 \times 10^{-3}}$
fNRI	6.85×10^{-3}

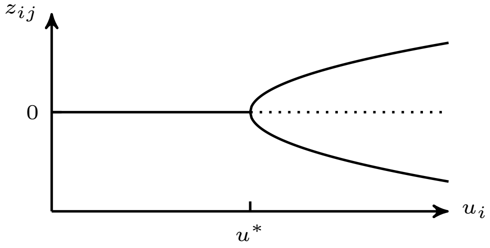


Figure 11: **Pitchfork bifurcation.** States are represented by z and u is the bifurcation parameter. When u is less than the critical value u^*, there is only one solution, $z = 0$. When u is larger than the critical value u^*, the solutions changes to three, $z \in \{\sqrt{u}, -\sqrt{u}, 0\}$. The stable (unstable) equilibrium solutions are illustrated with solid (dotted) lines.

$u > 0$, the solution is illustrated with a dotted line. The stable solutions $z \in \{\sqrt{u}, -\sqrt{u}\}$ are illustrated with solid lines.

In applications, the bifurcation parameter often corresponds to an external force. An example of a physical system whose solutions form a pitchfork bifurcation is a trailer being towed by a truck. In this case, the location of the center of gravity of the trailer is the bifurcation parameter and determines the stability of the truck-trailer system. A video showing the stable and unstable solutions of this physical system can be found at `https://www.youtube.com/watch?v=w9Dgxe584Ss`. A longer discussion of this system can be found in Horvath & Takacs (2022).

Separability of mutually exclusive categories. In the nonlinear opinion dynamics framework, the presence of mutually exclusive preferences simplifies the dynamics. Consider the case of two categories (i.e., $\mathcal{N}_\mathrm{o} = 2$). Preferences that are opposite in sign (i.e., $z_{i1} = -cz_{i2}$) and negatively correlated (i.e., $a^\mathrm{o}_{12}, a^\mathrm{o}_{21} < 0$) are mutually exclusive. Given these conditions, the belief term can be simplified to

$$\sum_{\substack{l=1 \\ l\neq j}}^{\mathcal{N}_\mathrm{o}} a^\mathrm{o}_{jl} z_{il} = a_{21} z_{i2} = -c a_{12} z_{i1}, \tag{32}$$

and the communication-belief term can be simplified to

$$\sum_{\substack{k=1 \\ k\neq i}}^{\mathcal{N}_\mathrm{a}} \sum_{\substack{l=1 \\ l\neq j}}^{\mathcal{N}_\mathrm{o}} a^\mathrm{a}_{ik} a^\mathrm{o}_{jl} z_{kl} = \sum_{\substack{k=1 \\ k\neq i}}^{\mathcal{N}_\mathrm{a}} a^\mathrm{a}_{ik} a^\mathrm{o}_{21} z_{k2} = \sum_{\substack{k=1 \\ k\neq i}}^{\mathcal{N}_\mathrm{a}} -a^\mathrm{a}_{ik} c a^\mathrm{o}_{12} z_{k1}. \tag{33}$$

With these simplification, the dynamics of z_{i1} in Equation (1) do not depend on z_{i2}. Defining the decoupled self-reinforcement term

$$\tilde{\alpha}_i = \alpha_{i1} - c a^o_{12}, \tag{34}$$

and communication matrix

$$\tilde{a}_{ik} = a^a_{i1} - c a^a_{i1} a^0_{12}, \tag{35}$$

Equation (1) can be simplified to Equation (2), that is,

$$\dot{z}_i = -d_i z_i + \mathcal{S}\left(u_i \left(\tilde{\alpha}_i z_i + \sum_{\substack{k=1 \\ k\neq i}}^{\mathcal{N}_\mathrm{a}} \tilde{a}_{ik} z_k \right)\right) + b_i. \tag{36}$$