

PAPER • OPEN ACCESS

Swarming network inference with importance clustering of relative interactions

To cite this article: Jason Hindes *et al* 2024 *J. Phys. Complex.* **5** 045009

View the [article online](#) for updates and enhancements.

You may also like

- [Self-organization in bacterial swarming: lessons from myxobacteria](#)
Yilin Wu, Yi Jiang, A Dale Kaiser et al.
- [Dynamical systems and complex networks: a Koopman operator perspective](#)
Stefan Klus and Nataša Djurdjevac Conrad
- [The ecological basis of morphogenesis: branching patterns in swarming colonies of bacteria](#)
Pan Deng, Laura de Vargas Roditi, Dave van Ditmarsch et al.



PAPER

OPEN ACCESS

RECEIVED
5 August 2024

REVISED
11 November 2024

ACCEPTED FOR PUBLICATION
6 December 2024

PUBLISHED
23 December 2024

Original Content from
this work may be used
under the terms of the
[Creative Commons
Attribution 4.0 licence](#).

Any further distribution
of this work must
maintain attribution to
the author(s) and the title
of the work, journal
citation and DOI.



Swarming network inference with importance clustering of relative interactions

Jason Hindes^{*} , Kevin Daley , George Stantchev and Ira B Schwartz

U.S. Naval Research Laboratory, Washington, DC 20375, United States of America

^{*} Author to whom any correspondence should be addressed.

E-mail: jason.m.hindes.civ@us.navy.mil

Keywords: swarming, networks, inference, clustering, dynamics, learning

Supplementary material for this article is available [online](#)

Abstract

Swarming is central to many problems in physics, biology, and engineering where collective motion and cooperation emerge through interactions of many agents. As a result, inferring network connections underlying swarms from time series data is an important problem with broad application. In this work, we propose a method based on phase-space regression of the observed dynamics with random forest models, from which relative interactions are clustered according to their Gini importance with respect to a given agent. Network connections are inferred by comparing how the statistics of the strongly and weakly important clusters overlap. Because the method entails fitting the dynamics and finding the most important interactions for each agent individually with general swarming assumptions, high accuracy can be maintained with relatively modest amounts of observation data and utilizing only a small number of generalizable hyperparameters across a variety of behavioral patterns, diverse physical ingredients, and heterogeneous network topologies.

1. Introduction

Swarms of mobile agents interacting through relatively simple rules are known to display complex, spatiotemporal, and coherent patterns [1]. Such systems are composed of many agents that consume energy in order to propel themselves in space, are out of equilibrium, and exhibit collective motion without central orchestration [2]. Natural examples include: colonies of bacteria [3, 4], swarms of insects [5–7], flocks of birds [8, 9], schools of fish [10, 11], and crowds of people [12, 13]. Much work has demonstrated how collective motion can emerge in swarms through physical and/or physically-inspired interactions, e.g. where temporal changes in the positions and velocities of agents are governed by state-dependent forces and inter-agent interactions that are often functions of relative coordinates [14–21]. In addition, swarm robotics has emerged as a field with growing interest, motivated by potential applications from cooperative exploration and mapping [22–24] to target seeking and tracking [25, 26], disaster recovery [27], resource allocation [28], and swarms for defense [29, 30].

Because of the ubiquity of swarming systems, inferring the dynamics and topology of unknown swarms from time series data is a problem of great interest. A typical approach is to extract the forces among agents in a swarm [31], assuming for instance zonal interaction models [32], and/or two and three-body interactions [33, 34]. Focusing specifically on topology, other promising techniques have leveraged general measures of similarity and data-driven clustering to infer networks underlying flocking behavior [35, 36], in which a swarm displays a high degree of polarization in its motion.

More generally, learning patterns of interactions in network-mediated dynamical systems is an area that has received much attention in recent years. A broad class of techniques approaches inference from an information-theoretic perspective, wherein direct links between agents are identified by a high degree of mutual information [37–39], time-delayed mutual information [40], information rates of change [41, 42], transfer entropy [43–45], and their generalizations. Other methods build graphical causal models with

time-delayed dependencies to infer interactions [46]. Another wide-ranging approach is to fit a network time series using a general regression method, wherein direct interactions are encoded in how the fit is performed. There are basis-function techniques, where a library of typical interactions [47] or complete orthogonal bases [48] are used to detect connections. Other methods fit a time series with well-known non parametric machine-learning frameworks, such as reservoir computing [49] or random forest regression [50, 51], where interactions correspond to non-zero Jacobian elements and features with high importance, respectively. In addition, methods applying deep neural networks to the problem of inference have also gained traction with high levels of accuracy [34, 52–55], though typically requiring large amounts of data and lacking interpretability. Lastly, most of the methods in the literature entail passive observation of dynamical systems, without actively probing a network to generate dynamics which may more efficiently enable inference. Several works have made progress along this latter line of research, e.g. [56, 57].

In this work, we introduce a network inference method that lies in between specialized, swarm-specific inference approaches and general network dynamics regression. Our technique is regression-based but designed for the following: to handle large swarming networks with sparse and heterogeneous topologies, to be robust in the presence of multiple swarming behaviors (emerging from nonlinear dynamics and noise), to provide interpretable results, and to be accurate with only moderate amounts of data. In particular, we perform regression of the temporal increments to the phase-space coordinates of agents within a swarm from time series data in order to infer which relative-interactions are important, but without specific functional forms using random-forest regression [58–60]. Then, we cluster interactions with Gaussian mixture models according to their importance to determine which represent direct interactions. Our work is different from other random-forest network inference techniques [50, 51], because it embeds relative-motion assumptions consistent with swarming physics, and uses unsupervised statistical clustering to process the regression importance, which inherently treats interactions with a given agent separately from the rest of the swarm.

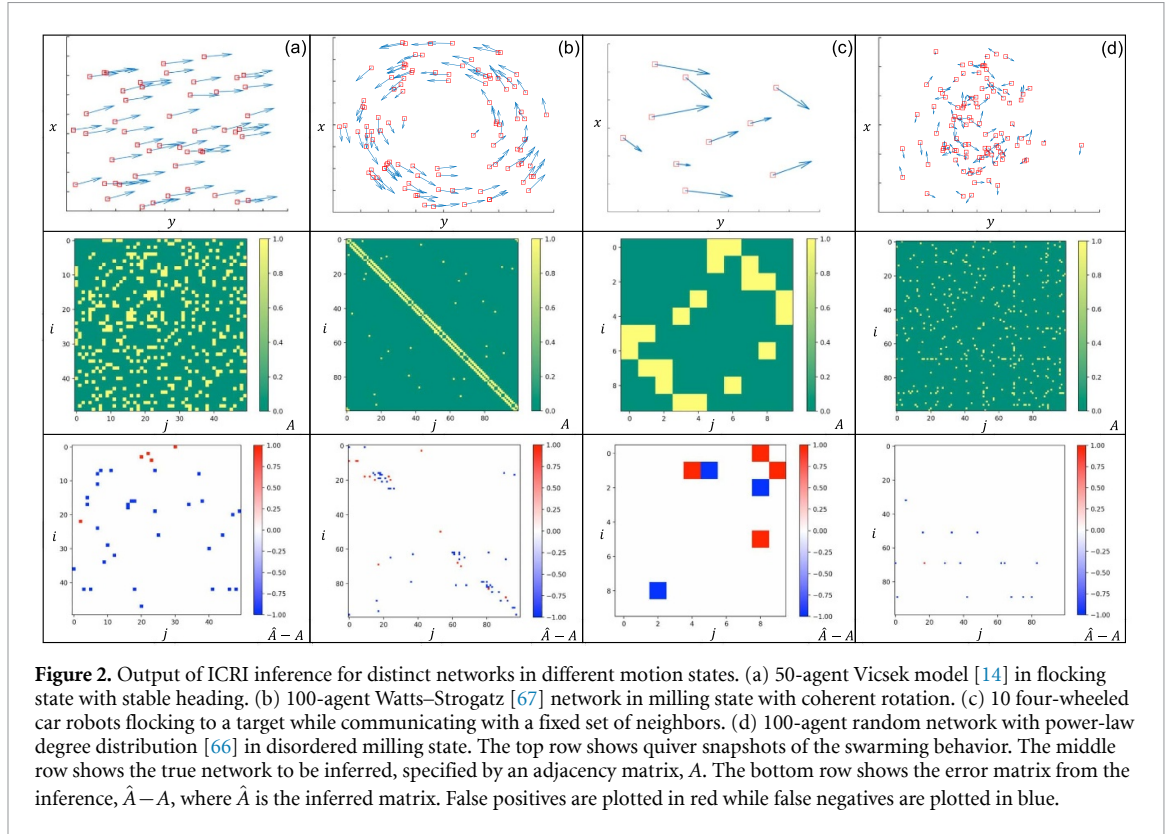
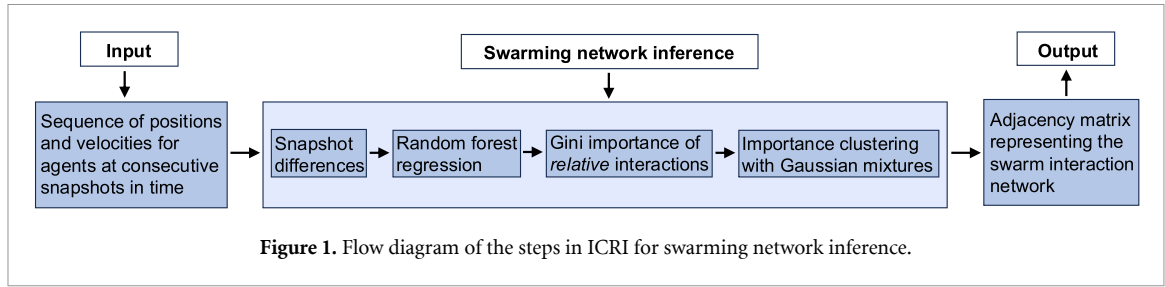
2. Summary of network inference from swarming dynamics

In general, the local dynamical rules for agents within a swarming system will consist of many ingredients with system-specific functional forms and parameters. Often, however, the dynamics contain some of the general physics of self-propulsion, damping, alignment, attraction, repulsion, etc in addition to any system-specific governing rules [1, 61], and external forces such as gravity. For example, at a minimum, most systems composed of mobile robots will have collision-avoidance controls, which can be expressed in simple terms as repulsive-force interactions [62–64].

We use the form of such physically-inspired interactions to extract information about unknown swarms, and specifically, infer the interaction topologies that influence their dynamics. In this work we create a dynamics regression and statistical procedure to determine the existence of pairwise interactions between agents in a swarm from snapshots of their trajectories. A flow diagram in figure 1 summarizes the approach, which takes as its input a position and velocity time series for swarming agents, and outputs an inferred adjacency matrix representing the interaction network among the agents. We use the acronym ICRI as a shorthand, throughout, and to emphasize its Importance Clustering of Relative Interactions basis. Section 3 elaborates further on technical details.

Because ICRI entails both fitting the dynamics through swarming assumptions and finding the most important interactions for each agent individually, high accuracy can be maintained in spite of topological heterogeneity, varying physical ingredients and parameters, diversity in the collective behavior observed, and relatively modest amounts of data. The flexibility of the approach results in superior performance as compared to information theoretic and similarity-clustering techniques, particularly for milling behaviors [65] and for swarming networks with high interaction densities.

As motivation, and to showcase the flexibility of our approach, we give an upfront demonstration of the ICRI network inference in a broad range of topologies and swarming systems. In figure 2, we show four example swarms, each in different states of motion. The top row contains typical snapshots of the positions and velocities for the swarms, while the middle and bottom rows plot the true adjacency matrices (specifying which agents directly interact) [66] and the error matrices from the inference, respectively. The example systems are quite different, notably in their topology: (a) is highly spatially correlated, (b) has significant topological clustering (the tendency for neighbors of agents to directly interact [67]), (c) is randomly constructed but topologically homogeneous, and (d) is random but heterogeneous, with the number of interactions per agent ranging over a factor of 30. In addition, the example (c) comes from a high-fidelity simulation of ten four-wheeled car robots, with only partially controlled dynamics, flocking to a target using



the software CoppeliaSim¹ [68]. Despite the significant differences among the examples, the ICRI method generalizes well and produces good accuracy as suggested by the sparse error matrices, even though it uses a small number of identical hyperparameters that are kept fixed across the different systems under analysis.

In the remainder of this paper we lay out the model systems used to test our method (section 2.1), present the details of how the ICRI network inference is performed (section 3), evaluate its performance as a function of swarm parameters (section 4), and compare to other techniques (section 5). The final section 6 discusses the results presented and provides commentary on generalizations and applications.

2.1. Model systems

To test our method, we primarily consider time series data generated from two general classes of swarming models. The first is a combination of standard models for self-propelled particles with second-order dynamics in their position vectors in two spatial dimensions (\mathbf{r}_i for the i th agent) [15–21, 29],

$$\ddot{\mathbf{r}}_i = (\alpha - \beta |\dot{\mathbf{r}}_i|^2) \dot{\mathbf{r}}_i + B_1 \sum_j A_{ij} \frac{\mathbf{r}_j - \mathbf{r}_i}{|\mathbf{r}_j - \mathbf{r}_i|} f_1(|\mathbf{r}_j - \mathbf{r}_i|) + B_2 \sum_j A_{ij} (\dot{\mathbf{r}}_j - \dot{\mathbf{r}}_i) f_2(|\mathbf{r}_j - \mathbf{r}_i|) + \boldsymbol{\xi}_i(t). \quad (1)$$

In equation (1), α is self-propulsion constant, β is a nonlinear damping constant, B_1 is a constant that sets the scale for position-dependent attractive and repulsive forces, B_2 is a constant that sets the scale for

¹ Agents are modeled as non-holonomic treaded robots that rely on skid-steering to produce motion. The control schema computes the required velocities for each tread in a global coordinate frame, based on a control function with a fixed communication network..

alignment-interactions, and $\xi_i(t)$ is a noise term. The attractive, repulsive, and alignment interactions are mediated through a fixed topology with adjacency matrix A , where $A_{ij} = 1$ if agents i and j interact and zero otherwise [66]. The above system can produce a variety of collective-motion behaviors including: flocking, milling, and rotational states depending on physical parameters, initial conditions, interaction networks, etc.²

The second model that we consider is the Vicsek model [14] where every agent (i) has a fixed speed (v) in two spatial dimensions, but with a dynamic heading angle (θ_i) that is updated in discrete time steps (Δt):

$$\begin{aligned}\theta_i(t + \Delta t) &= \langle \theta_j(t) \rangle_{|r_j - r_i| < d} + \xi_i(t), \\ \mathbf{r}_i(t + \Delta t) &= \mathbf{r}_i(t) + v(\cos(\theta_i(t)), \sin(\theta_i(t))) \Delta t.\end{aligned}\quad (2)$$

In equation (2), $\langle \theta_j(t) \rangle_{|r_j - r_i| < d}$ denotes the average heading of agents within a disk of radius d around agent i . Though the system has fewer physical ingredients (self-propulsion, alignment, finite-range sensing, and noise), the interaction topology is defined at every time step leading to time-dependent swarming networks in general. For comparison to ICRI network inference, we consider agents i and j to have $A_{ij} = 1$, if they interact through equation (2) for more than 50% of the snapshots in the time series data.

3. Method

As in other works, let us assume that we observe a swarm composed of agents $i = 1, 2, \dots, N$ over a series of time snapshots $p = 1, 2, \dots, P$, each with a state vector $\mathbf{q}_i^{(p)}$ at snapshot p . In particular, our primary focus will be on $\mathbf{q}_i = (x_i, y_i, \dot{x}_i, \dot{y}_i)$, where (x_i, y_i) is the center-of-mass position for agent i in two spatial dimensions, and where dots denote time derivatives. The time evolution for each agent i is given by a collection of increment vectors $\Delta \mathbf{q}_i^{(p)} = \mathbf{q}_i^{(p+1)} - \mathbf{q}_i^{(p)}$, for $p = 1, 2, \dots, P-1$. At each snapshot p , the increment vector $\Delta \mathbf{q}_i^{(p)}$ is assumed to depend approximately on the state vector $\mathbf{q}_i^{(p)}$ and the state vector differences $\mathbf{q}_i^{(p)} - \mathbf{q}_j^{(p)}$, for $j = 1, \dots, N, j \neq i$:

$$\Delta \mathbf{q}_i^{(p)} = \mathbf{F}_i \left(\mathbf{q}_i^{(p)}, \mathbf{q}_i^{(p)} - \mathbf{q}_1^{(p)}, \mathbf{q}_i^{(p)} - \mathbf{q}_2^{(p)}, \dots, \mathbf{q}_i^{(p)} - \mathbf{q}_N^{(p)} \right) + \xi(p), \quad (3)$$

where \mathbf{F}_i is a vector valued function that in general can be nonlinear and different for each agent i , and ξ is a noise source. We refer to \mathbf{F}_i as the *increment function* of agent i . Note that equation (3) is a discrete-time approximation for a general class of individual-based swarming models in continuous time [69, 70], which includes the models of section 2.1. and many others [1] as special limits. The dependence of \mathbf{F}_i on the state vector differences, rather than the state vectors themselves, implies that the swarm is assumed to evolve according to a *relative-interaction* dynamics among agents.

3.1. Regression of the increment functions with random forests

Assuming that the dynamics of the observed swarm is approximated by equation (3), for a collection of unknown \mathbf{F}_i , we infer the interaction topology from a sample time series by first fitting the increment functions with a nonlinear regression model that reflects the input/output structure of equation (3). We postulate that the interaction neighbors of a given agent i are associated with the model inputs of highest *feature importance* [50, 51]. One machine-learning model particularly well suited for this setting is random-forest regression, because it allows for fitting a general nonlinear \mathbf{F}_i , it has a built-in definition of importance for its input variables (namely *Gini importance* [58–60], and variants thereof, such as *permutation importance* [71]), it requires modest amounts of data for training (especially for the high-dimensional systems that we consider), and it can be designed to avoid over-fitting in the presence of noise.

In brief, random forests are ensembles of regression trees, each of which is a sequential, optimal binary partitioning of its training data [58–60]. At each ‘leaf’ in a tree, a single feature variable and threshold value are chosen to partition the dataset into two branches. The predicted value within each branch is the data average within the branch. The feature and threshold are selected to minimize the mean-squared error of this prediction. For every tree in a random forest, the partitioning is performed a fixed number of times, corresponding to a maximum depth, d , which controls the complexity. A schematic diagram depicting the motion regression from time series snapshots using random forests with relative interactions is shown in the upper two panels of figures 3(a) and (b). In panel (a), we show how the time snapshots are used to extract the

² Unless stated otherwise, we take $f_1(r) = (C_a/l_a) \exp(-r/l_a) - (C_r/l_r) \exp(-r/l_r)$, which describes short range attractive and repulsive forces between agents with length scales l_a and l_r , respectively [16]. Similarly, we take $f_2 = \exp(-r/l_v)$ where l_v is a sensing length scale for alignment.

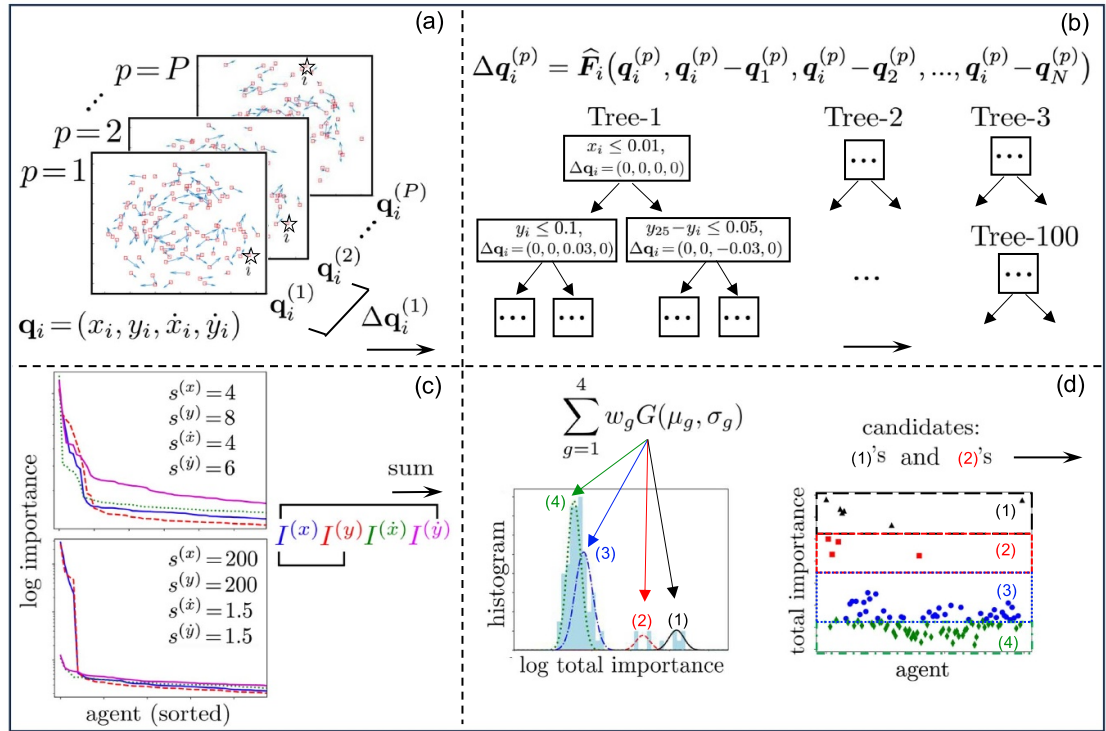


Figure 3. Schematic of the steps in ICRI for swarming network inference. For every i th agent in a swarm: (a) Compute the increments to i 's state space, measuring the change between snapshots in the time series. (b) Fit the increment function for i with a random forest model that takes as its arguments the state-space vector for i , and the difference between i 's vector and the vectors for all other agents. (c) Output the (Gini) importance from the random forest for all state-space variables, and form a single total importance score for every agent relative to i according to equations (4) and (5). (d) Divide the agents into four types by performing a mixed-Gaussian clustering of the log total importance relative to agent i . Agents in the two clusters with the highest means are candidates for interactions, determined by a final set of statistical tests equations (7) and (8). Steps (a)–(d) are repeated for all agents $i = 1, 2, \dots, N$.

increments for agents, which comprise the training outputs. In panel (b), we illustrate how the increment function for agent i is fit with regression trees, and plot example leaves in the first tree with selection variables, thresholds, and predicted increments.

After fitting each increment function \mathbf{F}_i with a random-forest model $\hat{\mathbf{F}}_i$, we infer the interaction neighbors of agent i by analyzing whose relative state vectors have the highest Gini importance [58–60, 72] for $\hat{\mathbf{F}}_i$. In our analysis, the Gini importance with respect to agent i is the fractional reduction in the mean-squared error of $\hat{\mathbf{F}}_i$ (in predicting the $\{\Delta \mathbf{q}_i^{(1)}, \Delta \mathbf{q}_i^{(2)}, \dots, \Delta \mathbf{q}_i^{(P-1)}\}$) produced by each relative interaction. As explained above, since trees fit data based on a series of single-variable conditions, the Gini importance of a given interaction to $\hat{\mathbf{F}}_i$ is computed by simply adding up the mean-squared error reduction every time an interaction appears in the random forest. Our numerical implementation leverages the Python library scikit-learn [73], and in particular its random forest regression methods, which compute Gini importance following [58, 59, 72]. See supplementary material for additional details on the numerical implementation. Hence, $\hat{\mathbf{F}}_i$ is associated with importance scores for each possible relative interaction. In particular when $\mathbf{q}_i = (x_i, y_i, \dot{x}_i, \dot{y}_i)$, there are four Gini scores for agent $j \neq i$ with respect to agent i , denoted $I_{ij}^{(x)}$, $I_{ij}^{(y)}$, $I_{ij}^{(\dot{x})}$, and $I_{ij}^{(\dot{y})}$, corresponding to the phase-space differences $x_i - x_j$, $y_i - y_j$, $\dot{x}_i - \dot{x}_j$, and $\dot{y}_i - \dot{y}_j$, respectively.

3.2. Total importance estimates

In order to find the interactions of agent i , we aim to build an overall or total importance score for every other agent $j \neq i$ from the collection: $I_{ij}^{(x)}$, $I_{ij}^{(y)}$, $I_{ij}^{(\dot{x})}$, and $I_{ij}^{(\dot{y})}$. A straightforward method would be to add up position and velocity scores for j . However, depending on the state of motion, the physical interactions, how noise enters, etc Gini scores for some of the coordinates can be associated with larger uncertainty, particularly $I_{ij}^{(\dot{x})}$ and $I_{ij}^{(\dot{y})}$. To see this, in figure 3(c) we show $I_{ij}^{(x)}$, $I_{ij}^{(y)}$, $I_{ij}^{(\dot{x})}$, and $I_{ij}^{(\dot{y})}$ for all agents with respect to a typical agent i in a flocking (top) and milling (bottom) swarm generated from equation (1). Note that the importance scores are plotted on log-scale on the vertical axis. For each coordinate type, agent scores are sorted from highest to lowest Gini importance with respect to agent i . For the flocking example, scores for

different coordinates fall off similarly and rapidly. However, for the milling example, the velocity scores do not. In fact, they are relatively flat across the swarm—reaching a level of background above the position scores and implying no distinction between agents. Possible conclusions are that agent i interacts with, effectively, the entire swarm, or that $I_{ij}^{(x)}$ and $I_{ij}^{(y)}$ are associated with a high degree of noise and uncertainty. Based on the assumption that each agent interacts (sparsely) with a relatively small number of neighbors compared to the total N , we conclude the latter. Hence, when forming a total importance score, we want to consider only coordinate types whose Gini scores decay sufficiently fast (e.g. $I_{ij}^{(x)}$ and $I_{ij}^{(y)}$ in figure 3(c, lower)), and discount scores that are flat across agents.

To measure how fast each type of Gini score decays, for every agent i we compare the highest score to the m th highest score, where $m = \lfloor N\rho \rfloor$ and $0 < \rho \leq 1$ is chosen to be roughly on the order of the network sparsity. Specifically, for a finite set S let $\max(S, m)$ denote the m th largest element. As a measure of decay rate, we compute the ratio of the highest Gini score to the m th highest Gini score and take the average value over all agents $1 \leq i \leq N$ for each coordinate type. For instance, for the x -coordinate:

$$s^{(x)} = \left\langle \frac{\max \left(\left\{ I_{ij}^{(x)} \right\}_{j=1}^N, 1 \right)}{\max \left(\left\{ I_{ij}^{(x)} \right\}_{j=1}^N, m \right)} \right\rangle_i \quad (4)$$

where $\langle \cdot \rangle_i$ denotes averaging over the elements of a set indexed by i . Note that a larger $s^{(x)}$ implies a clearer distinction between low and high $\{I_{ij}^{(x)}\}_{j=1}^N$ scores. The ratios $s^{(y)}$, $s^{(x)}$, and $s^{(y)}$ are defined analogously to equation (4). Example values for $s^{(x)}$, $s^{(y)}$, $s^{(x)}$, and $s^{(y)}$ are given in figure 3(c) for their respective swarms.

As mentioned, when creating a total importance estimate, we only want Gini scores associated with coordinates that exhibit a more pronounced decay across all agents. Therefore, we find the coordinate type that has the most rapid decay, namely $s_{\max} = \max(\{s^{(x)}, s^{(y)}, s^{(x)}, s^{(y)}\}, 1)$ and use it as a benchmark. Then, we add Gini scores from other coordinates only if they are within a certain tolerance, $0 < r < 1$, of s_{\max} . Doing so, gives the total importance estimate for agent j with respect to agent i :

$$I_{ij} = H\left(\frac{s^{(x)}}{s_{\max}} - r\right) I_{ij}^{(x)} + H\left(\frac{s^{(y)}}{s_{\max}} - r\right) I_{ij}^{(y)} + H\left(\frac{s^{(x)}}{s_{\max}} - r\right) I_{ij}^{(x)} + H\left(\frac{s^{(y)}}{s_{\max}} - r\right) I_{ij}^{(y)}, \quad (5)$$

where $H(x)$ is the Heaviside function defined as $H(x) = 1$ if $x \geq 0$ and zero otherwise.

3.3. Gaussian mixture model for clustering the regression importance

The final step of ICRI is turning the total importance value I_{ij} defined in equation (5) into an inferred adjacency matrix entry \hat{A}_{ij} . Since the scores can range over several orders of magnitude, in general, we process $\ln I_{ij}$. In sparse, heterogeneous swarms with noisy dynamics, the distribution of total importance scores with respect to every agent can be quite different. For instance, the amount of background noise, the importance scores of the most important interactions, etc can all vary from agent to agent, especially if the number of interactions per agent is heterogeneous across the network. For these reasons, it is best to analyze the scores with respect to each agent (i) separately. Statistical clustering allows us to process scores in a common way, based on similar statistical patterns, but without a large number of hyperparameters. In this work, we perform the desired clustering with Gaussian mixture models [73, 74], which allow for uncertainty quantification and more flexibility as compared to other methods.

For each agent i we assume a Gaussian mixture model for the set $\{I_{i1}, I_{i2}, \dots, I_{iN}\}$ with 4 components for the following reasons. First, to find the true interactions of agent i , we want to remove background noise associated with unimportant agents, which nominally comprise a majority of the swarm. We do so by defining a background type in the Gaussian mixture that we denote type (4). Outside this background type, the remaining agents can be split into three types according to the likelihood of true interaction with agent i : (1) *yes*, (2) *maybe*, and (3) *no*. We postulate that type (1) agents truly interact with agent i , types (3) and (4) agents do not, and type (2) agents require further analysis.

Given the four types, the Gaussian mixture clustering entails fitting the distribution of $\{I_{i1}, I_{i2}, \dots, I_{iN}\}$ with a weighted sum of four parametric Gaussian distributions

$$\ln I_{ij} \sim \sum_{g=1}^4 w_{i,g} \mathcal{N}(\mu_{i,g}, \sigma_{i,g}). \quad (6)$$

In equation (6), \mathcal{N} denotes a normal distribution with mean $\mu_{i,g}$ and standard deviation $\sigma_{i,g}$, with a relative frequency $w_{i,g}$ for importance type $g \in \{1, 2, 3, 4\}$ relative to agent i . Once the fit is performed, agent j is

classified according to which Gaussian component maximizes the likelihood for $\ln I_{ij}$ to have been drawn from it. An example of the importance clustering with respect to a typical agent is plotted in figure 3(d). Note that the four types are arranged according to decreasing means: $\mu_{i,1} \geq \mu_{i,2} \geq \mu_{i,3} \geq \mu_{i,4}$.

Once we have classified all agents, we perform further analysis for type (2) agents in order to decide whether they represent true interactions. To do so, we apply two additional statistical criteria. First, the scores for type (2) should be well-separated from those for type (3). We build a quantitative criterion expressing this condition by stipulating that typical fluctuations in (2) below its mean should not overlap with typical fluctuations in (3) above its mean (expressed in terms of the standard deviations)³:

$$\mu_{i,2} - z\sigma_{i,2} \geq \mu_{i,3} + z\sigma_{i,3}, \quad (7)$$

where z is a hyperparameter. Second, the scores for type (2) should not be too far separated from scores of type (1). Since, by assumption, the characteristic distance between interactions and non-interactions is $\mu_{i,1} - \mu_{i,3}$, if type (2) is representative of the former, then $\mu_{i,1} - \mu_{i,2}$ should be a significantly smaller fraction of this distance,

$$\mu_{i,1} - \mu_{i,2} \leq f(\mu_{i,1} - \mu_{i,3}), \quad (8)$$

where $0 \leq f \leq 1$ is another hyperparameter. Finally, since the parameters for the Gaussian mixture model are determined based on stochastic iterative optimization of the probability likelihood [73, 74], we perform the classification based on equation (6) over 20 independent trials and check the conditions, equations (7) and (8), for each trial. If agent j is determined to have an interaction with agent i in n_t out of the 20 trials, then we set $\hat{A}_{ij} = 1$; otherwise $\hat{A}_{ij} = 0$. The final hyperparameter for ICRI is n_t . See supplementary material for additional details on the importance clustering implementation.

3.4. Hyperparameter calibration

The ICRI method depends on four primary hyperparameters: the maximum depth of trees d , the overlap parameter z between importance types (2) and (3), the distance parameter f for importance types (1) and (2), and the stochastic threshold parameter n_t . The additional importance sparsity and tolerance parameters, ρ and r , produce less performance sensitivity, and we leave them fixed at $\rho = 0.1$ and $r = 0.3$. Our goal is to select suitable values for the remaining hyperparameters that give satisfactory performance for different networks in various swarming states, over a range of physical parameters and models, and then keep those values fixed throughout the method evaluations discussed in the remainder of the paper. In this way we try to control the tendency to overfit a flexible machine learning method to a specific dataset.

To track the performance, we adopt as our primary measure the \mathcal{F}_1 score, which is the harmonic mean of precision and recall for binary classification [75]. The \mathcal{F}_1 score is useful, since it accounts for errors from both false ‘positives’ and false ‘negatives’. For network inference with a binary adjacency matrix let:

$$\begin{aligned} T^+ &= \{(i, j) \mid A_{ij} = 1, \hat{A}_{ij} = 1\} \\ F^+ &= \{(i, j) \mid A_{ij} = 0, \hat{A}_{ij} = 1\} \\ F^- &= \{(i, j) \mid A_{ij} = 1, \hat{A}_{ij} = 0\} \end{aligned}$$

denote respectively the set of indices of the true positive, false positive, and false negative predicted links. Then the \mathcal{F}_1 score of a given network inference model is defined as:

$$\mathcal{F}_1 = \frac{2\#T^+}{2\#T^+ + \#F^+ + \#F^-} \quad (9)$$

where $\#$ denotes the cardinality of the respective set [75]. By construction, the \mathcal{F}_1 score ranges from 0 to 1. Our aim is to achieve scores that are primarily in the range considered excellent, or $\mathcal{F}_1 > 0.9$, without fine tuning.

In figure 4 we plot the dependence of the \mathcal{F}_1 score on each of the four primary hyperparameters in panels (a)–(d). For each panel, red labels correspond to a network with flocking behavior, whereas blue labels correspond to a network with milling behavior. For each series we deliberately choose different networks, physical parameters, initial conditions, and models so that the results are robust in a variety of systems. A

³ Because the clusters are only finite samples from Gaussian distributions, often with small numbers, traditional hypothesis tests are generally less accurate for inference than equations (7) and (8).

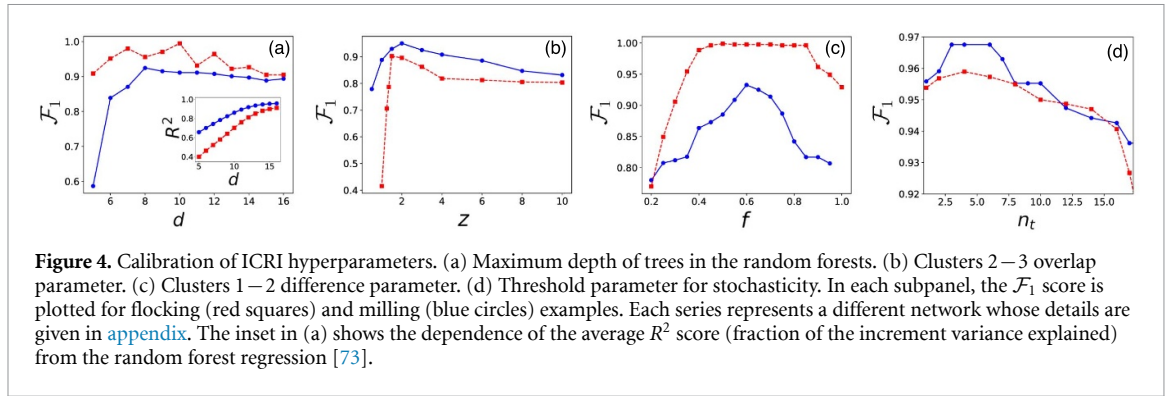


Table 1. Hyperparameters for ICRI swarming network inference.

parameter	description	value
d	maximum depth of trees in random forests	10
z	importance clusters 2–3 overlap	1.5
f	importance clusters 1–2 difference	0.6
n_t	threshold for stochasticity	5
ρ	importance sparsity	0.1
r	importance decay-rate tolerance	0.3

discussion of the eight, significantly different swarming networks used in figure 4 can be found in [appendix](#). We point out that within each subplot, the other hyperparameters are set to the operational values used throughout the rest of the paper.

Optimal values for a given swarm could be found by maximizing \mathcal{F}_1 with respect to all hyperparameters. However, in principle such an approach results in different parameters for each example. Here, we select nominal values based on figure 4 that are stably in the target range for all 8 swarms, but do not rely on strict optimization. The fixed values for the remainder of this work (and figure 2) are listed in table 1.

4. Performance with varying swarm parameters

Once we fix the hyperparameters, we can explore how ICRI performs as we change the properties of the swarming system. In particular, we want to verify its robustness under a variety of physical parameters, network topologies, and swarming behaviors. Table 2 summarizes the swarm simulation parameters used (unless stated otherwise).

4.1. External parameter dependence

In this section we test performance on external parameters, such as the number of snapshots in the time series (P), and the intensity of noise (D). Both the small-data regime and the high-noise limit are expected to degrade the performance of ICRI, and hence it is important to quantify changes in accuracy as these parameters are swept. In figure 5 we plot the \mathcal{F}_1 score as a function of P and D for several examples. As in figure 4, we follow the convention that flocking networks are plotted in red and milling networks are plotted in blue. The flocking swarm is an 8-regular random graph and the milling swarm is a Watts–Strogatz network (with $\langle k \rangle = 10$ and 10% random interactions [67]), both with 100 agents. The dynamics are generated from equation (1), where the noise is white and spatially uncorrelated Gaussian with variance $2D$.

First, in figure 5(a) we observe a monotonic increase in performance with P that is quite rapid until $P \sim 10^4$, after which gains are limited. Given the 400 degrees of freedom in the swarming examples, the nominal point $P \sim 10^4$ is modest compared to other methods, particularly those involving deep neural networks. We can gain a sense of why the performance changes as we increase P by looking at example error matrices like the ones shown in figure 5(c). As we move from left to right, we increase P from 10^3 to 3×10^3 . In the left example there are many rows of the inferred adjacency matrix where agents are predicted to have a very large number of interactions, as indicated by red horizontal bands in the plotted error matrix. In fact, the predicted mean number of interactions per agent is $\langle \hat{k} \rangle \approx 12$, which is 50% higher than the true value. On the other hand, when $P = 3 \times 10^3$ the bands disappear and the mean number of interactions is correctly predicted to be $\langle \hat{k} \rangle \approx 8$. The tests in figure 5(a) demonstrate a general finding, which we see for many examples in this work, that our method requires moderate amounts of data to get satisfactory performance, both with different topologies and collective behaviors.

Table 2. Swarm simulation parameters.

parameter	description	value(s)
P	number of time snapshots	10^4
D	noise intensity	10^{-2}
Δt	time-difference between snapshots	0.05
N	number of agents	8–200
$\langle k \rangle$	average number of interactions per agent	2–20
σ_k	standard deviation of the number of interactions per agent	0–10

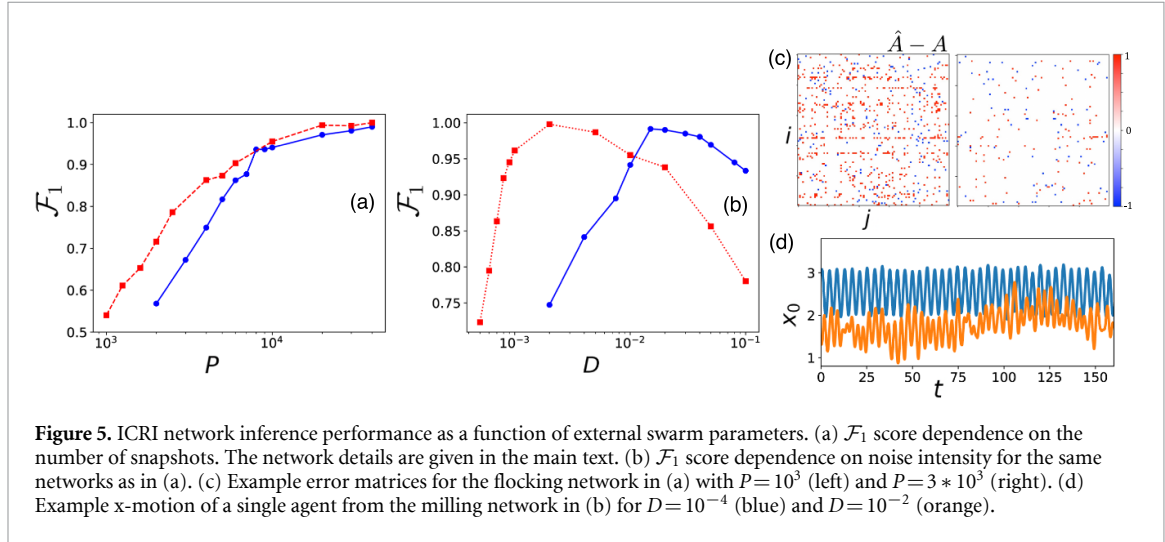


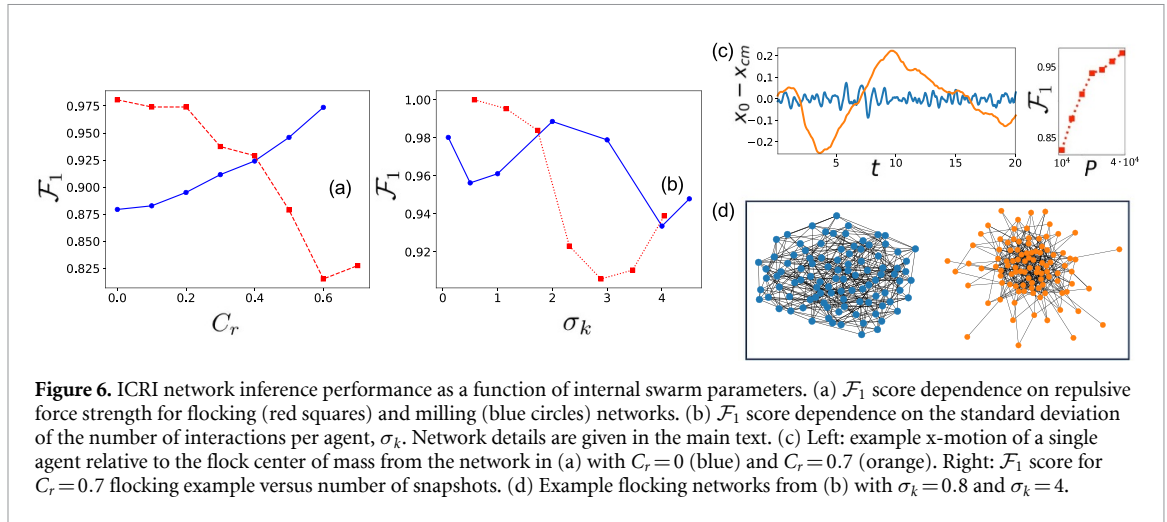
Figure 5. ICRI network inference performance as a function of external swarm parameters. (a) \mathcal{F}_1 score dependence on the number of snapshots. The network details are given in the main text. (b) \mathcal{F}_1 score dependence on noise intensity for the same networks as in (a). (c) Example error matrices for the flocking network in (a) with $P = 10^3$ (left) and $P = 3 \times 10^3$ (right). (d) Example x -motion of a single agent from the milling network in (b) for $D = 10^{-4}$ (blue) and $D = 10^{-2}$ (orange).

Second, the dependence on noise intensity is more subtle. In particular, the \mathcal{F}_1 scores increase for small values of noise in figure 5(b), before decreasing when the noise becomes too large. Similar patterns have been found in other works [49]. A qualitative explanation is that, if the noise is too small, the swarming system can reside near a much lower dimensional *deterministic* attractor compared to N , which does not carry enough information to extract the network by regression. On the other hand, if the noise is too large, the swarming dynamics tends to be driven by noise alone, and not the deterministic contribution to the dynamics that includes interactions, as in models equations (1) and (2). To see this, we give a suggestive example of the dynamics in figure 5(d), where the x -coordinate of a single agent is plotted for small (blue) and moderate (orange) noise intensities. In the former, the time series is approximately a simple sine wave. Consequently, even though the number of snapshots in the time series is $P = 10^4$, the data is in fact not very expressive, since it merely repeats the same pattern after a relatively short window. Viewed in light of the periodicity, the poor performance is unsurprising. In contrast, for the moderate noise example the time series is more complex, with a broad Fourier spectrum indicative of aperiodic motion [65]. For this latter example we achieve the target accuracy, $\mathcal{F}_1 > 0.9$. The examples in figures 5(b) and (d) are representative of the general trend that if a swarm is observed near a stable spatiotemporal pattern, noise can increase the inference performance of ICRI by generating more expressive data. The need for sufficiently high-dimensional time series data to achieve target levels of performance is an issue that we return to in section 5.

4.2. Internal parameter dependence

Beyond external parameters, we can test how ICRI performs as internal swarm parameters change. Two important internal parameters are the repulsive force strength and the topological heterogeneity of the underlying network. The first is significant because repulsive forces tend to produce spatial separation and negative phase-space correlations in the relative motions between interacting agents. The second is significant because heterogeneity makes it difficult to process the interactions of topologically dissimilar agents within a common framework. Several examples from the model system equation (1) are given in figure 6.

First, following the structure of figure 5, we plot the \mathcal{F}_1 score as a function of the repulsive force strength C_r in figure 6(a) for an Erdos–Rényi network with 25 agents and $\langle k \rangle = 4$ [66]. For both flocking (red) and milling (blue) the inference performance lies within the target range with $\mathcal{F}_1 > 0.9$ for most values of C_r . However, we can see that the flocking inference appears to show a degradation in accuracy with high levels of repulsion. We can understand the cause by comparing the dynamics when the repulsion is $C_r = 0$ and



$C_r = 0.7$. An example is shown in figure 6(c, left) where we plot the x -motion of a single agent for the two cases in blue and orange, respectively.

For the former, the dynamics fluctuate rapidly, indicative of an expressive time series. In the latter, high repulsion slows the fluctuations around the swarm center of mass. In fact, only 25 time windows of the sort plotted in figure 6(c, left) are included in the entire dataset for inference when $P = 10^4$. As such, the degradation in performance for large repulsion has more to do with having insufficient time series data. In fact, doubling the dataset size to $P = 2 * 10^4$ increases the performance to $\mathcal{F}_1 > 0.9$, or comparable to the weak-repulsion limit. The increased accuracy with more time series data is demonstrated in figure 6(c, right). Overall, the tests in figure 6 demonstrate a general finding that our method is able to accommodate swarms with different physical parameters, without a significant change in performance. The reason for robustness is our reliance on the swarming assumptions of equation (3) which do not change as specific model parameters are altered.

Second, to study the inference performance with topological heterogeneity, we hold the mean number of interactions constant $\langle k \rangle = 8$, and vary the standard deviation using tunable networks. Two example series are shown in figure 6(b) for flocking (red) and milling (blue) swarms. Each network is a configuration model random graph [66] with 100 agents. The flocking networks have uniform degree distributions, and the milling networks have Gaussians, both discretized to integer numbers of interactions. We can see that for both swarming network classes, the performance fluctuates but is stable over significant heterogeneity with $\mathcal{F}_1 > 0.9$ throughout. To help visualize the difference in the networks as we vary the distribution of numbers of interactions per agent, we depict the limiting networks for flocking in figure 6(d). In particular, the more heterogeneous network in orange has a complex topology with an inner core of highly interacting agents and a sparse periphery where agents have as few as one interaction. The examples in figure 6(b), as well as figure 2, demonstrate that our method generalizes well and maintains target levels of performance in topologically heterogeneous systems, even without the need for hyperparameter re-calibration. Underlying the robustness to heterogeneity is the fact that the ICRI method fits the swarming dynamics, assigns importances, and classifies the importances separately for each agent.

4.3. Homing pigeon example

So far, we have tested ICRI on time series data generated from model systems and robotics simulations with partially controlled dynamics. In this section, we explore performance using a biological dataset of homing pigeons [76]. The data consist of positions and velocities for pigeons in a flock in three dimensions with time resolution of 0.2 s using GPS tracking. An example trajectory from [76] is shown in figure 7(a) in the horizontal plane (for ‘homing flight number 2’). By studying time-dependent correlations in the normalized velocities, the authors found that the pigeons tended to organize themselves into hierarchical patterns with directional interactions, in which pigeons were more likely to mimic the headings of certain individuals and not others.

In particular, one can define a time-delayed correlation of the normalized velocities:

$$\mathcal{C}_{ij}(\tau) = \langle \mathbf{v}_i(t) \cdot \mathbf{v}_j(t + \tau) \rangle, \quad (10)$$

given a time-delay τ and normalized velocity \mathbf{v}_i for agent i . One then looks for τ^* that maximizes the correlation between i and j . If the τ^* is negative, a plausible meaning is that agent i tends to mimic agent j ,

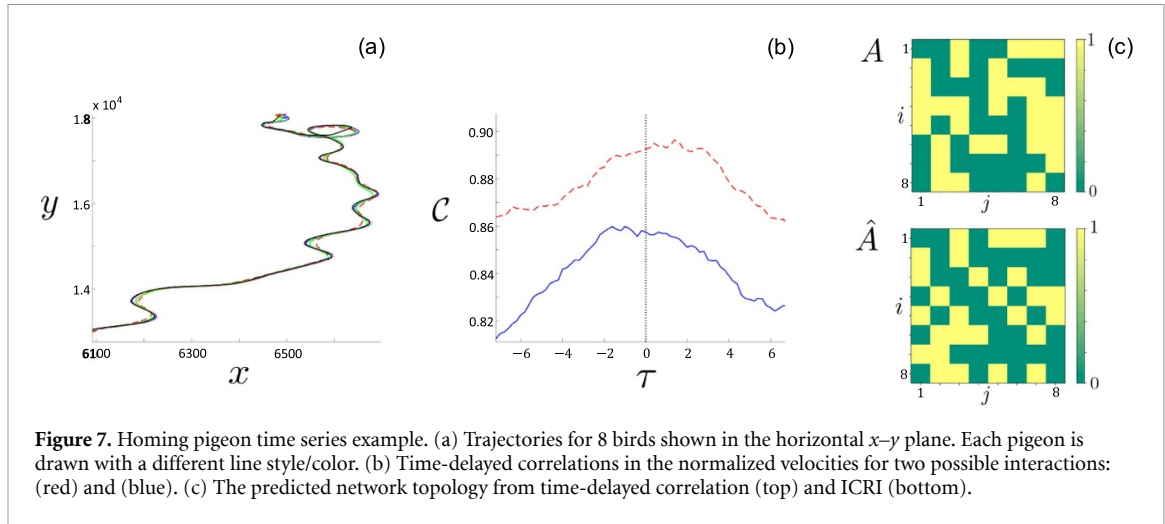


Figure 7. Homing pigeon time series example. (a) Trajectories for 8 birds shown in the horizontal x - y plane. Each pigeon is drawn with a different line style/color. (b) Time-delayed correlations in the normalized velocities for two possible interactions: (red) and (blue). (c) The predicted network topology from time-delayed correlation (top) and ICRI (bottom).

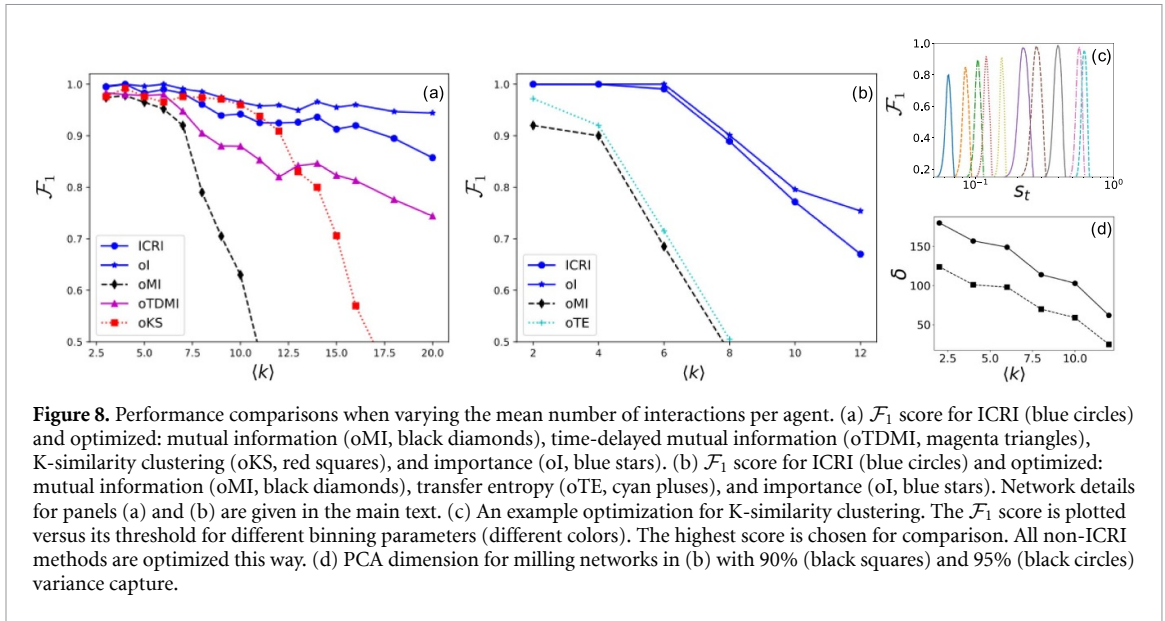
and thus agent j would be considered a leader of agent i . For agents $j \neq i$ with negative (or zero) τ_{ij}^* with respect to agent i we can assign an $A_{ij} = 1$, as long as the correlation is above some threshold (i.e. $C_{ij}(\tau^*) > 0.5$). Doing so, one finds networks that are qualitatively robust under different motion scenarios [76]. An example of two correlation functions with respect to the same pigeon i are shown for two candidate neighbors in figure 7(b). For the blue neighbor $\tau^* < 0$, while for the red neighborhood, $\tau^* > 0$. Consequently, the former can be reasonably assigned $A_{ij} = 1$, and the latter $A_{ij} = 0$.

Here we repeat the analysis of the directed-correlation network (from homing flight number 2), and compare to the output of ICRI. As with the former analysis, the data input consists of the normalized velocities for each pigeon in the dataset. Note that in this example, there are a total of eight pigeons. The network derived from the correlational analysis is shown in the upper panel of figure 7(c). The output of ICRI is shown in the lower panel. Using the correlational network as ground truth, the ICRI inference demonstrates fair agreement, with $\mathcal{F}_1 = 0.73$. The performance is comparable to other machine-learning methods tested on homing-pigeon networks [35]. However, ICRI does not rely on the approximation of undirected interactions, and correctly predicts a non-symmetric A . In addition, both the correlational and ICRI analyses agree roughly on the number of interactions (around 30), and that pigeon 7 has the lowest (or tied for the lowest) in-degree, meaning that it mimics the fewest pigeons. Similarly, pigeons 7 and 3 have the highest out-degrees, meaning that they tend to be mimicked more by others, and are therefore higher in the leadership hierarchy; both networks give similar results for pigeons that are lower in the leadership hierarchy. Of course, we point out that ICRI is designed to capture close-in-time dependencies (extracting how important a given agent is in changing the current state of another from one time step to the next). In contrast, connectivity derived from relatively long correlation times may detect secondary interactions as opposed to direct ones, which can explain some of the apparent discrepancy between the two analyses. Nevertheless, the network predicted by ICRI is broadly consistent with the findings in [76], and could be useful for further uncovering other interactions implicit in the swarming behavior beyond heading mimicry.

5. Comparison to other methods

Finally, we compare the performance of ICRI to other network inference techniques. The techniques compared against in this work are common and relatively straightforward: pairwise mutual information [37–39], pairwise time-delay mutual information [40], pairwise transfer entropy [43–45], and K-similarity clustering [35]. Note that we tested several other inference methods [46, 47], but their performance was relatively poor, at least for the swarming dynamics of interest here and without significant tailoring, so we excluded them from the study. In general, each chosen comparison technique has a binning parameter (or several binning parameters) for unsupervised clustering, a choice of state variables to cluster⁴, a time-delay (for time-delay mutual information and transfer entropy), and a threshold parameter. When comparing to ICRI, we vary all of the free parameters and choose the best for comparison. Ultimately, each technique requires taking a measure of interaction between agents and turning the measure into an inferred

⁴ For all comparison methods we tested different combinations of state variables for every agent i , including the full state-space vector, \mathbf{q}_i . However, in general, using the full \mathbf{q}_i gave worse performance. We found the best performance for the information-theoretic techniques by clustering velocity data, whereas for K-similarity clustering, the best results were obtained by clustering position data.



adjacency matrix through a threshold. This approach inherently works best when agents are at least approximately topologically similar, e.g. in their number of interactions. Therefore, to achieve the highest compatibility among the techniques under comparison (in addition to parameter optimization) we test on topologically homogeneous swarms. Lastly, a fundamental difficulty for any network inference method is that, as the number of interactions an agent has becomes large, it becomes challenging to separate direct from indirect interactions. Therefore, we would like to compare performance as a function of the number of interactions.

In figure 8 we show example comparisons for flocking (a) and milling (b) swarms versus the average number of interactions per agent. The flocking networks are k -regular configuration model random graphs [66]. The milling networks are Watts–Strogatz networks with 10% random interactions [67]. All of the series have $N=100$ agents. In addition to the comparison techniques which are optimized, we plot the ICRI method with blue circles (which is not optimized), and a variation with blue stars in which a single optimal global threshold is chosen for the total importance matrix I , defined through equation (5), as with the comparison techniques.

Several key trends are apparent in figures 8(a) and (b). First, for small numbers of interactions $\langle k \rangle \lesssim 8$, our method performs as well or better than the other methods *even without optimization* (figure 8(c)); recall that identical hyperparameters are used for all blue circle points. Second, all of the other techniques degrade precipitously when the number of interactions become moderately large, $\langle k \rangle \gtrsim 8$. In particular, our method is able to more effectively distinguish between direct and indirect interactions for a wider range of $\langle k \rangle$. Third, even our method eventually reaches only fair performance when $\langle k \rangle$ becomes too large. In figure 8 the flocking example remains in the target performance range $\mathcal{F}_1 > 0.9$ with $\langle k \rangle = 20$, but the milling example moves out of range when $\langle k \rangle \geq 10$.

A partial explanation for the degradation of performance with high $\langle k \rangle$ was hinted at in section 4. Namely, as $\langle k \rangle$ becomes large, the effective dimensionality of the time series decays. This is particularly the case for the milling swarms, figure 8(b). We can quantify the reduction in the effective dimensionality by tracking the principle component analysis (PCA) [77] dimension of the swarm time series. In this application, the PCA entails finding an optimal linear basis for the state space that captures some fraction of the temporal variance [65]. Two examples are shown in figure 8(d) for two fractions of the time series variance captured for the milling swarm. We can see that as $\langle k \rangle$ increases, the effective PCA dimension (δ) of the milling data eventually becomes significantly smaller than the number of agents $N=100$. We suggest $\delta \lesssim N$ as a crossover point at which sparse network inference techniques based on regression will have difficulty. For learning the network underlying swarming systems with such low dynamical complexity, one would likely need to generate transient behavior with higher values of δ [47, 56, 57].

6. Conclusions

Collective motion of many coupled mobile agents is ubiquitous and includes a wide variety of spatiotemporal behaviors across many scales in physics, biology, and engineering. Yet, in large unknown

swarming systems it is difficult to extract patterns of interactions, especially when individual agents can have many degrees of freedom in several spatial dimensions. Despite the different space and time scales of known swarming systems, from active particles to birds and robots, many swarms are driven in part by basic physical and/or physically-inspired interactions and forces. Such ingredients typically entail increments to the phase-space degrees of freedom for each agent locally in space and time based on state-dependent forces and controls for the agents, as well as interaction forces that are functions of relative coordinates with respect to the rest of the swarm.

In this work, we developed a method that leverages these insights for swarming network inference. First, random-forest regression models were fit to the increment data for each agent in order to extract the reduction in prediction error due to the inclusion of every possible *relative* interaction. Then, an importance estimate was obtained for every agent with respect to a given agent. Finally, the importance scores were fit to a Gaussian mixture model and thresholded using the statistics of the clusters to separate direct interactions from indirect ones and background noise. Because of its physics-informed structure and the fact that clusters were formed and analyzed for each agent separately, the proposed method was able to achieve high accuracy compared to other techniques in spite of complicating factors such as topological heterogeneity, diversity in the swarming behavior, physical parameter variation, and high interaction densities.

Our approach represents a step-forward towards inferring large, sparse, and heterogeneous swarming networks, particularly when there is an absence of highly polarized collective motion. Hence, we expect suitable variations to be useful for inferring links in other large dynamical systems with nonlinear interactions and noise, for cases between sparse and dense data regimes, where some partial physical assumptions are known, and where interpretability for informing subsequent science is desirable. Important examples include networks of coupled phase oscillators, which are used to model, for instance, high-voltage electric power grids, where the dynamics often evolve with relative state-space interactions as well [78]. Yet, limitations for our method point toward improvements and possible generalizations. A key direction would entail more precise uncertainty quantification of importance and a probabilistic formulation for inferring interactions more generally. This could be achieved, for instance, by using more flexible measures of feature importance, such as permutation importance [71], and/or developing a non-parametric Bayesian framework [79] for clustering importance where an optimal number of clusters is inferred separately for each agent. Another major direction would address optimal perturbations to a swarm for generating sufficiently expressive data and enabling more accurate network inference, particularly for dense swarming networks. Such perturbations could involve probing a swarm through collision with external agents, where the subsequent transients could be used to infer interactions. Addressing these issues, among others, will form the basis of future work.

Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

Acknowledgments

JH and IBS were supported by the U.S. Naval Research Laboratory funding (N0001423WX00008), and the Office of Naval Research (N0001424WX01006) and (N0001424WX00813). KD was supported through the National Research Council postdoctoral fellowship program. GS supported by the Office of Naval Research (N0001424WX01006). The authors thank Chinthan Prasad for performing and explaining the robotics simulations which support this manuscript.

Appendix

We briefly discuss the swarming networks used in the hyperparameter calibration, section 3.4, and figure 4. First, in figure 4(a) we plot the inference performance as function of the maximum depth of the regression trees for a 50 agent tree network with a power-law degree distribution (red) and a 100 agent Watts–Strogatz network [67] with average degree $\langle k \rangle = 10$ and with a fraction of interactions (10% in this case), chosen at random (blue). Both examples are generated from equation (1). In either case, the \mathcal{F}_1 score is near the target range ($\mathcal{F}_1 > 0.9$) when the tree depth of the random forest regressor is at or close to $d = 10$. In contrast, if we consider the accuracy of the increment-function fit on the training data, we expect a monotonic increase with d . This is shown in the inset panel of figure 4(a), where we plot the R^2 (fraction of the variance captured by the random forest regression) versus d . As we might expect, we find that for values of d which are too

large, the models tend to overfit to the time series and generally have lower accuracy for inference. The value $d = 10$ is roughly in the range where underfitting and overfitting are avoided.

Second, in figure 4(b) we plot the inference performance as function of z for two configuration model networks [66] generated from the swarming equation (1). The flocking swarm is a 100-agent bimodal network where 90 agents have 3 interactions while 10 agents have 15 interactions [21]. The milling swarm is a 100-agent power-law degree distributed network, $k^{-2.5}$ [66]. For both examples we find \mathcal{F}_1 scores near the target range when $1 < z < 2$. Recall from equation (7) that if $z = 0$ we effectively count all interactions classified as type (2) to be direct, whereas if z is large, no type (2) interactions are considered direct. We pick $z = 1.5$ roughly between these two limits.

Third, in figure 4(c) we plot the inference performance as a function of f for two random networks generated from equation (1). The flocking swarm is a configuration model network with 100 agents where all agents have 5 interactions (called 5-regular) [66]. The underlying network of the milling swarm is a 200-agent Barabási–Albert graph (with $m = 3$ [66]). The limiting dependencies with f from equation (8) are similar to z .

Finally in figure 4(d) we plot the inference performance as a function of n_t , recalling that in order for $\hat{A}_{ij} = 1$, I_{ij} must be classified as a type (1) or type (2) interaction with equations (7) and (8) having been satisfied in n_t out of 20 trials. The flocking network is a 50-agent Vicsek swarm modelled with equation (2). Since the Vicsek swarming network is time dependent, we build an effective adjacency matrix to be inferred in which agents i and j have $A_{ij} = 1$ if they interact in at least half of the P snapshots generated from equation (2). The milling network is a 25-agent Erdős–Rényi graph, in which the mean number of interactions is $\langle k \rangle = 4$ [66]. For both examples, we can see that \mathcal{F}_1 has a maxima near $n_t = 5$, with only a modest reduction in performance for more restrictive thresholding.

ORCID iDs

Jason Hindes  <https://orcid.org/0000-0001-7051-7978>

Kevin Daley  <https://orcid.org/0000-0003-2652-1675>

Ira B Schwartz  <https://orcid.org/0000-0001-5999-2524>

References

- [1] Vicsek T and Zafeiris A 2012 Collective motion *Phys. Rep.* **517** 71–140
- [2] Psimen L 2021 *Active Matter Within and Around Us: From Self-Propelled Particles to Flocks and Living Forms* 1st edn (Springer)
- [3] Copeland M F and Weibel D B 2009 Bacterial swarming: a model system for studying dynamic self-assembly *Soft Matter* **5** 1174–87
- [4] Be'er A and Ariel G 2019 A statistical physics view of swarming bacteria *Mov. Ecol.* **7** 9
- [5] Theraulaz G et al 2002 Spatial patterns in ant colonies *Proc. Natl Acad. Sci. USA* **99** 9645–9
- [6] Sinhuber M, van der Vaart K, Feng Y, Reynolds A M and Ouellette N T 2021 An equation of state for insect swarms *Sci. Rep.* **11** 3773
- [7] Topaz C M, D'Orsogna M R, Edelstein-Keshet L and Bernoff A J 2012 Locust dynamics: behavioral phase change and swarming *PLOS Comput. Biol.* **8** 1–11
- [8] Young G F, Scardovi L, Cavagna A, Giardina I, Leonard N E and Bergstrom C T 2013 Starling flock networks manage uncertainty in consensus at low cost *PLOS Comput. Biol.* **9** 1–7, 01
- [9] Ballerini M et al 2007 Interaction ruling animal collective behavior depends on topological rather than metric distance: evidence from a field study *Proc. Natl Acad. Sci.* **105** 1232–7
- [10] Tunström K, Katz Y, Ioannou C C, Huepe C, Lutz M J and Couzin I D 2013 Collective states, multistability and transitional behavior in schooling fish *PLoS Comput. Biol.* **9** 1–11
- [11] Calovi D S, Lopez U, Ngo S, Sire C, Chaté H and Theraulaz G 2014 Swarming, schooling, milling: phase diagram of a data-driven fish school model *New J. Phys.* **16** 015026
- [12] Helbing D, Johansson A and Zein Al-Abideen H 2007 Dynamics of crowd disasters: an empirical study *Phys. Rev. E* **75** 046109
- [13] Chraïbi M, Kemloh V, Schadschneider A and Seyfried A Force-based models of pedestrian dynamics *Netw. Heterog. Media.* **6** 425–42
- [14] Vicsek T, Czirók A, Ben-Jacob E, Cohen I and Shochet O 1995 Novel type of phase transition in a system of self-driven particles *Phys. Rev. Lett.* **75** 1226–9
- [15] Levine H, Rappel W-J and Cohen I 2000 Self-organization in systems of self-propelled particles *Phys. Rev. E* **63** 017101
- [16] D'Orsogna M R, Chuang Y L, Bertozzi A L and Chayes L S 2006 Self-propelled particles with soft-core interactions: Patterns, stability and collapse *Phys. Rev. Lett.* **96** 104302
- [17] Cucker F and Smale S 2007 Emergent behavior in flocks *IEEE Trans. Autom. Control* **52** 852–62
- [18] Choi Y-P, Ha S-Y and Li Z 2017 *Emergent Dynamics of the Cucker–Smale Flocking Model and Its Variants* (Springer) pp 299–331
- [19] Hindes J, Edwards V, Szwajkowska Kasraie K, Stantchev G and Schwartz I B 2021 Swarm shedding in networks of self-propelled agents *Sci. Rep.* **11** 13544
- [20] Hindes J, Edwards V, Kamimoto S, Triandaf I and Schwartz I B 2020 Unstable modes and bistability in delay-coupled swarms *Phys. Rev. E* **101** 042202
- [21] Hindes J, Szwajkowska K and Schwartz I B 2016 Hybrid dynamics in delay-coupled swarms with mothership networks *Phys. Rev. E* **94** 032306
- [22] Huang X, Arvin F, West C, Watson S and Lennox B 2019 Exploration in extreme environments with swarm robotic system 2019 *IEEE Int. Conf. on Mechatronics (ICM)* vol 1 pp 193–8

- [23] Pires A G, Rezeck P A F, Chaves R A, Macharet D G and Chaimowicz L 2021 Cooperative localization and mapping with robotic swarms *J. Intell. Robot. Syst.* **102** 47
- [24] Lynch K M, Schwartz I B, Yang P and Freeman R A 2008 Decentralized environmental modeling by mobile sensor networks *IEEE Trans. Robot.* **24** 710–24
- [25] Deepak B B V L and Parhi D R 2013 Target seeking behaviour of an intelligent mobile robot using advanced particle swarm optimization *2013 Int. Conf. on Control, Automation, Robotics and Embedded Systems (CARE)* pp 1–6
- [26] John Amala Arokia Nathan R J, Kurmi I and Bimber O 2023 Drone swarm strategy for the detection and tracking of occluded targets in complex environments *Commun. Eng.* **2** 55
- [27] Gladence L, Anu V, Anderson A, Stanley I, Fernando J A and Revathy S 2021 Swarm intelligence in disaster recovery *2021 5th Int. Conf. on Intelligent Computing and Control Systems (ICICCS)* pp 1–8
- [28] Li H, Feng C, Ehrhard H, Shen Y, Cobos B, Zhang F, Elamvazhuthi K, Berman S, Haberland M and Bertozzi A L 2017 Decentralized stochastic control of robotic swarm density: Theory, simulation and experiment *2017 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)* pp 4341–7
- [29] Walton C, Kaminer I, Gong Q, Clark A H and Tsatsanifos T 2022 Defense against adversarial swarms with parameter uncertainty *Sensors* **22** 4773
- [30] Hindes J, Edwards V, Hsieh M A and Schwartz I B 2021 Critical transition for colliding swarms *Phys. Rev. E* **103** 062602
- [31] Lukeman R, Li Y-X and Edelstein-Keshet L 2010 Inferring individual rules from collective behavior *Proc. Natl Acad. Sci.* **107** 12576–80
- [32] Ling H, Mclvor G E, van der Vaart K, Vaughan R T, Thornton A and Ouellette N T 2019 Local interactions and their group-level consequences in flocking jackdaws *Proc. R. Soc. B* **286** 20190865
- [33] Katz Y, Tunström K, Ioannou C C, Huepe C and Couzin I D 2011 Inferring the structure and dynamics of interactions in schooling fish *Proc. Natl Acad. Sci.* **108** 18720–5
- [34] Ruiz-García M, Barriuso C M G, Alexander L C, Aarts D G A L, Ghiringhelli L M and Valeriani C 2024 Discovering dynamic laws from observations: the case of self-propelled, interacting colloids *Phys. Rev. E* **109** 064611
- [35] Gu K, Duan X, Qi M and Yan L 2023 A global relative similarity for inferring interactions of multi-agent systems *Complex Intell. Syst.* **9** 1671–86
- [36] Liang J, Qi M, Gu K, Liang Y, Zhang Z and Duan X 2022 The structure inference of flocking systems based on the trajectories *Chaos* **32** 101103
- [37] Gavel S, Singh Raghuvanshi A and Tiwari S 2022 Maximum correlation based mutual information scheme for intrusion detection in the data networks *Expert Syst. Appl.* **189** 116089
- [38] Ross B C and Marinazzo D 2014 Mutual information between discrete and continuous data sets *PLoS One* **9** 1–5
- [39] Tirabassi G, Sevilla-Escoboza R, Buldú J M and Masoller C 2015 Inferring the connectivity of coupled oscillators from time-series statistical similarity analysis *Sci. Rep.* **5** 10829
- [40] Li S, Xiao Y, Zhou D and Cai D 2018 Causal inference in nonlinear systems: Granger causality versus time-delayed mutual information *Phys. Rev. E* **97** 052216
- [41] Bianco-Martínez E, Rubido N, Antonopoulos C G and Baptista M S 2016 Successful network inference from time-series data using mutual information rate *Chaos* **26** 043102
- [42] Antonopoulos C G 2023 Network inference combining mutual information rate and statistical tests *Commun. Nonlinear Sci. Numer. Simul.* **116** 106896
- [43] Schreiber T 2000 Measuring information transfer *Phys. Rev. Lett.* **85** 461–4
- [44] Novelli L and Lizier J T 2021 Inferring network properties from time series using transfer entropy and mutual information: validation of multivariate versus bivariate approaches *Netw. Neurosci.* **5** 373–404
- [45] Novelli L, Wollstadt P, Mediano P, Wibral M and Lizier J T 2019 Large-scale directed network inference with multivariate transfer entropy and hierarchical statistical testing *Netw. Neurosci.* **3** 827–47
- [46] Runge J, Nowack P, Kretschmer M, Flaxman S and Sejdinovic D 2019 Detecting and quantifying causal associations in large nonlinear time series datasets *Sci. Adv.* **5** eaau4996
- [47] Casadiego J, Nitzan M, Hallerberg S and Timme M 2017 Model-free inference of direct network interactions from nonlinear collective dynamics *Nat. Commun.* **8** 2192
- [48] Wang S, Herzog E D, Kiss I Z, Schwartz W J, Bloch G, Sebek M, Granados-Fuentes D, Wang L and Li J-S 2018 Inferring dynamic topology for decoding spatiotemporal structures in complex heterogeneous networks *Proc. Natl Acad. Sci.* **115** 9300
- [49] Banerjee A, Hart J D, Roy R and Ott E 2021 Machine learning link inference of noisy delay-coupled networks with optoelectronic experimental tests *Phys. Rev. X* **11** 031014
- [50] Leguia M G, Levnajić Z, Todorovski L and Ženko B 2019 Reconstructing dynamical networks via feature ranking *Chaos* **29** 093107
- [51] Leng S, Xu Z and Ma H 2019 Reconstructing directional causal networks with random forest: causality meeting machine learning *Chaos* **29** 093130
- [52] Frolov N, Maksimenko V, Lüttjohann A, Koronovskii A and Hramov A 2019 Feed-forward artificial neural network provides data-driven inference of functional connectivity *Chaos* **29** 091101
- [53] Murphy C, Laurence E and Allard A 2021 Deep learning of contagion dynamics on complex networks *Nat. Commun.* **12** 4720
- [54] Ding X, Kong L-W, Zhang H-F and Lai Y-C 2024 Deep-learning reconstruction of complex dynamical networks from incomplete data *Chaos* **34** 043115
- [55] Koch J, Chen Z, Tuor A, Drgona J and Vrabie D 2023 Structural inference of networked dynamical systems with universal differential equations *Chaos* **33** 023103
- [56] Delabays R and Tyloo M 2021 Network inference using sinusoidal probing *IFAC-PapersOnLine* **54** 696–700
- [57] Timme M 2007 Revealing network connectivity from response dynamics *Phys. Rev. Lett.* **98** 224101
- [58] Breiman L 2001 Random forests *Mach. Learn.* **45** 5–32
- [59] Geurts P, Ernst D and Wehenkel L 2006 Extremely randomized trees *Mach. Learn.* **63** 3–42
- [60] Biau G and Scornet E 2016 A random forest guided tour *Test* **25** 197–227
- [61] Fu Y, Yu H, Zhang X, Malgaretti P, Kishore V and Wang W 2022 Microscopic swarms: from active matter physics to biomedical and environmental applications *Micromachines* **13** 295
- [62] Faruk Keceli E and Ceccarelli M 2015 *Mobile Robots for Dynamic Environments* 1st edn (American Society of Mechanical Engineers)
- [63] Melchiorre M, Sabatino Scimmi L, Salamina L, Mauro S and Pastorelli S 2022 Robot collision avoidance based on artificial potential field with local attractors *Proc. 19th Int. Conf. on Informatics in Control, Automation and Robotics - ICINCO (INSTICC, SciTePress)* pp 340–50

- [64] Merheb A-R, Gazi V and Sezer-Uzol N 2016 Implementation studies of robot swarm navigation using potential functions and panel methods *IEEE/ASME Trans. Mechatronics* **21** 2556–67
- [65] Kamimoto S, Hindes J and Schwartz I B 2023 The chaotic milling behaviors of interacting swarms after collision *Chaos* **33** 081106
- [66] Newman M E J 2010 *Networks: an Introduction* (Oxford University Press)
- [67] Watts D J and Strogatz S H 1998 Collective dynamics of ‘small-world’ networks *Nature* **393** 440–2
- [68] Rohmer E, Singh S P N and Freese M 2013 Coppeliasim (formerly v-rep): a versatile and scalable robot simulation framework *Proc. Int. Conf. on Intelligent Robots and Systems (IROS)* (<https://doi.org/10.1109/IROS.2013.6696520>)
- [69] Bolley F, Ca nizo J A and Carrillo J A 2011 Stochastic mean-field limit: non-lipschitz forces & swarming *Math. Models Methods Appl. Sci.* **21** 2179–210
- [70] Bolley F, Ca nizo J A and Carrillo J A 2012 Mean-field limit for the stochastic vicsek model *Appl. Math. Lett.* **25** 339–43
- [71] Altmann A, Tološi L, Sander O and Lengauer T 2010 Permutation importance: a corrected feature importance measure *Bioinformatics* **26** 1340–7
- [72] Stone C J Olshen Leo Breiman R A and Friedman J 1984 *Classification and Regression Trees* (Chapman and Hall/CRC)
- [73] Pedregosa F et al 2011 Scikit-learn: machine learning in Python *J. Mach. Learn. Res.* **12** 2825–30
- [74] Deisenroth M P, Faisal A A and Soon Ong C 2020 *Mathematics for Machine Learning* (Cambridge University Press)
- [75] Japkowicz N 2013 *Assessment Metrics for Imbalanced Learning* (Wiley) ch 8, pp 187–206
- [76] Nagy M’e, Ákos Z, Biro D and Vicsek T 2010 Hierarchical group dynamics in pigeon flocks *Nature* **464** 890–3
- [77] Brunton S L and Nathan Kutz J 2022 *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems and Control* 2nd edn (Cambridge University Press)
- [78] Witthaut D, Hellmann F, Kurths J, Kettemann S, Meyer-Ortmanns H and Timme M 2022 Collective nonlinear dynamics and self-organization in decentralized power grids *Rev. Mod. Phys.* **94** 015005
- [79] Orbanz P and Whye Teh Y 2010 *Bayesian Nonparametric Models* (Springer) pp 81–89