Erschienen in: Computational Particle Mechanics ; 1 (2014), 4. - S. 425-440
https://dx.doi.org/10.1007/s40571-014-0030-7

Coarse-grained variables for particle-based models: diffusion maps and animal swarming simulations

Ping Liu · Hannah R. Safford · Iain D. Couzin · Ioannis G. Kevrekidis

Abstract As microscopic (e.g. atomistic, stochastic, agent-based, particle-based) simulations become increasingly prevalent in the modeling of complex systems, so does the need to systematically coarse-grain the information they provide. Before even starting to formulate relevant coarse-grained equations, we need to determine the right macroscopic *observables*—the right variables in terms of which emergent behavior will be described. This paper illustrates the use of data mining (and, in particular, diffusion maps, a nonlinear manifold learning technique) in coarse-graining the dynamics of a particle-based model of animal swarming. Our computational data-driven coarse-graining approach extracts two coarse (collective) variables from the detailed particle-based simulations, and helps formulate a low-dimensional stochastic differential equation in terms of these two collective variables; this allows the efficient quantification of the interplay of "informed" and "naive" individuals in the collective swarm dynamics. We also present a brief exploration of swarm breakup and use data-mining in an attempt to identify useful predictors for it. In our discussion of the scope and limitations of the approach we focus on the key step of selecting an informative metric, allowing us to usefully compare different particle swarm configurations.

Keywords Particle-based models · Coarse-graining · Data mining · Swarming

P. Liu · H. R. Safford
Department of Chemical and Biological Engineering,
Princeton University, Princeton, NJ 08544, USA
e-mail: pingliu@princeton.edu

Present address:
P. Liu
Department of Molecular, Cellular and Developmental Biology,
Yale University, West Haven, CT 06516, USA

Present address:
H. R. Safford
Woodrow Wilson School, Princeton University,
Princeton, NJ 08544, USA
e-mail: hsafford@princeton.edu

I. D. Couzin
Department of Ecology and Evolutionary Biology,
Princeton University, Princeton, NJ 08540, USA
e-mail: icouzin@princeton.edu

I. G. Kevrekidis (✉)
Department of Chemical and Biological Engineering, Program in Applied and Computational Mathematics, Princeton University,
Princeton, NJ 08544, USA
e-mail: yannis@princeton.edu

1 Introduction

A persistent feature of particle-based models is the emergence of macroscopic, collective behavior from the interactions of individual particles (e.g. catalyst particles in a fluidized bed, cells in a chemotactic gradient, individual animals in a population) between themselves and with their environment. To macroscopically usefully describe the behavior of such models we should be able to deduce system-level, macroscopic evolution rules from the microscopic particle motion/interaction rules. Before we can envision the derivation of such macroscopic equations, however, we must first identify the relevant coarse level descriptors: the "right variables", in terms of which the macroscopic dynamics will be described, and in terms of which the macroscopic models will be formulated. For many traditional physical problems, these variables are well established from experience (often backed up by mathematical arguments). Typically, they are the first few, low-order moments of the particle distribution, e.g. concentrations and temperature for chemical kinetics, or

density and momentum fields for isothermal Newtonian fluid mechanics, with the addition of stresses for non-Newtonian flows. As we work with increasingly novel and complex problems, however, for which extensive experience and intuition is lacking, the selection of good coarse-grained variables becomes a nontrivial task, even a bottleneck, in the path towards useful macroscopic modeling.

Aiming to address this limitation, and to propose a systematic approach in identifying useful coarse-grained descriptors, we illustrate in this paper a data mining approach that helps identify the relevant coarse variables from fine scale simulation data. In particular, starting with a particle-based (individual agent-based) animal swarming model developed by Couzin et al. [1], we use diffusion maps (DMAP, a nonlinear manifold learning technique) [2–6] to capture key features of the collective motion of the swarm. This, in turn, helps succintly describe and understand how a group of socially interacting animals switches its direction of collective motion. Based on the identified DMAP coarse variables, we then construct a reduced (effective) stochastic differential equation (SDE) model which allows us to efficiently quantify the stochastic collective direction switching behavior of the animal group.

The paper is organized as follows: In Sect. 2, the particle-based animal swarming model is described. In Sect. 3, a brief description of the DMAP approach and its underpinnings is given. Section 4 discusses the implementation of the DMAP approach and the (crucial) selection of an appropriate pairwise similarity measure between nearby particle simulation snapshots (nearby swarm configurations). In Sect. 5 we extract a two dimensional effective description of the collective animal motion; Sect. 6, exploits the reduced SDE effective model and the DMAP coarse variables to compute the mean escape time between rare dynamic events: in our case this is the time (on average) it takes for the group to switch between preferred directions of motion. In Sect. 8 we briefly explore the phenomenon of (computational) swarm breakup, and try to identify latent coarse variables that can serve as useful predictors for it. We summarize and conclude with a discussion of the scope of our approach, its links to multiscale computation and its limitations in Sect. 9.

2 The animal swarming model

We will illustrate, through a nontrivial, informative example, how data mining can complement and enhance the extraction of useful macroscopic dynamic information from detailed, particle-based computations. The illustrative example is an animal swarming model, and the relevant dynamic information in this context is the distribution of times between rare events (motion switching between two preferred directions); yet, even though the particular model and particular task are of interest in themselves, what we hope to convey is a computer-assisted "enabling approach" to detailed modeling, and a sense of its scope and limitations.

The particle-based animal swarming model we study was developed by Couzin et al. [1]. We use this model to study how collective decisions are made in a *heterogeneous* group when two sub-groups of *leaders* (informed individuals) have different opinions (information) about the location of the resources (e.g. food). We consider a group of N individuals moving on the plane. Each individual is characterized at a given moment in time by a position vector $\mathbf{c}_i(t)$, a direction vector $\mathbf{d}_i(t)$, and has speed s (we assume all individuals have the same constant speed). Individuals attempt to maintain a minimum distance r_p between themselves and other individuals; this avoidance is the highest priority in each individual's rule of motion. It is reflected in the model by defining a *repulsion zone*, Ω_p, which is a local neighborhood of radius r_p. The desired direction of travel $\mathbf{d}_i$ is updated as follows:

$$\mathbf{d}_i(t+\Delta t) = -\sum_{j\in\Omega_p, j\neq i} \frac{\mathbf{c}_j(t)-\mathbf{c}_i(t)}{\left|\mathbf{c}_j(t)-\mathbf{c}_i(t)\right|}. \tag{1}$$

If no other individuals are detected within the repulsion zone, then the individual is attracted towards, and aligns its direction of travel with, other individuals in a larger neighborhood called the *attraction zone*, Ω_a. This region has radius r_a. The resulting desired direction of travel is given by,

$$\mathbf{d}_i(t+\Delta t) = \sum_{j\in\Omega_a, j\neq i} \frac{\mathbf{c}_j(t)-\mathbf{c}_i(t)}{\left|\mathbf{c}_j(t)-\mathbf{c}_i(t)\right|} + \sum_{j\in\Omega_a} \frac{\mathbf{d_j(t)}}{\left|\mathbf{d_j(t)}\right|}. \tag{2}$$

Here $\mathbf{d}_i(t+\Delta t)$ is converted to the corresponding unit vector $\hat{\mathbf{d}}_i(t+\Delta t) = \mathbf{d}_i(t+\Delta t)/\left|\mathbf{d}_i(t+\Delta t)\right|$. The model is comprised of three different subgroups of particles; we denote them as Group A, Group B and Group U respectively. Group A and Group B are the "informed individuals", and they have different *preferred directions* (simulated as unit vectors $\mathbf{g}_A$ and $\mathbf{g}_B$ respectively) representing, for example, the directions to two different known food resources; here these directions will be diametrically opposed - "up" and "down" on the plane. All other individuals (Group U) are *naive* and have no preference to move in any particular direction. Informed individuals balance their preferred directions and the social interactions with others with a weighting factor ω,

$$\mathbf{d}_i'(t+\Delta t) = \frac{\hat{\mathbf{d}}_i(t+\Delta t)+\omega\mathbf{g}_i}{\left|\hat{\mathbf{d}}_i(t+\Delta t)+\omega\mathbf{g}_i\right|}. \tag{3}$$

In this study we let Group A agents have preferred directions pointing upward (i.e. $\mathbf{g}_A = (0,1)^T$ in the 2D physical space), and let Group B agents have preferred directions pointing downward ($\mathbf{g}_B = (0,-1)^T$). When the weighting

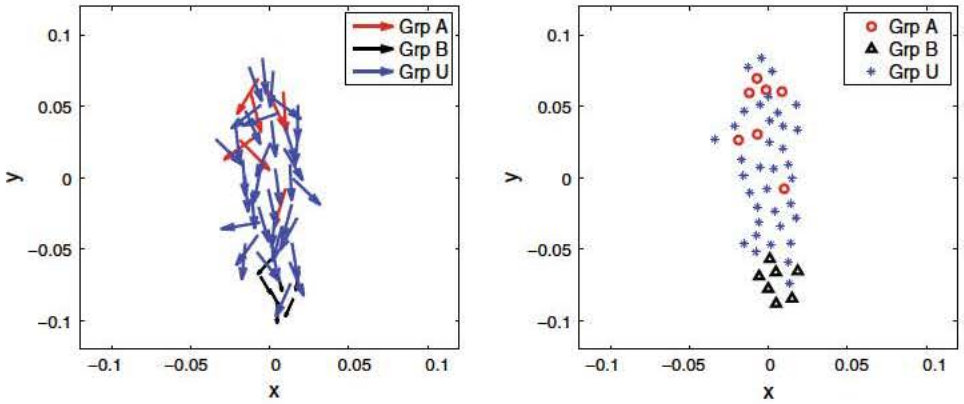

Fig. 1 Sample snapshots of simulation results for the particle-based animal swarming model. **a** The directions of travel for the individuals. The *red arrows* denote directions for Group A individuals (the informed agents whose preferred directions are pointing upward). The *black arrows* denote directions of Group B individuals (the informed agents whose preferred directions are pointing downward). The *blue arrows* denote the directions of Group U individuals (the uninformed agents). **b** The relative positions of each individual (after subtracting the center of mass of the entire group). The *red circles* denote the relative positions for Group A agents; the *black triangles* denote the relative positions for Group B agents; and the *blue asterisks* denote the relative positions for the uninformed agents. (Color figure online)

factor ω is relatively small, the informed individuals have weak desire to move toward their preferred directions, and so does the entire group. As ω increases, the level of influence by the preferred directions increases, while influence due to the social interactions decreases. At intermediate values of ω, the group is observed to randomly switch between collective upward motion and collective downward motion. As ω further increases, the group can break into two parts, each consisting of one informed group plus a few uninformed individuals. In this study we focus on such "intermediate" values of ω, and study how the group switches its direction of collective motion.

To account for an individual's error in perception and motion, the direction of motion $\mathbf{d}_i'(t+\Delta t)$ is rotated by a Gaussian random angle with a standard deviation σ, resulting in $\mathbf{d}_i''(t+\Delta t)$. There is also a constraint on the rate of turning (the turning angle per unit time η). If the angle between $\mathbf{d}_i''(t+\Delta t)$ and the individual's previous direction of motion $\mathbf{d}_i''(t)$ is less than $\eta\Delta t$, then the desired direction of motion at time $t+\Delta t$ is set to $\mathbf{d}_i''(t+\Delta t)$. Otherwise, $\mathbf{d}_i''(t+\Delta t)$ is obtained by turning $\mathbf{d}_i'(t+\Delta t)$ by an angle $\eta\Delta t$ towards $\mathbf{d}_i''(t+\Delta t)$. Finally, the position of each individual is updated as follows:

$$\mathbf{c}_i(t+\Delta t) = \mathbf{c}_i(t) + \mathbf{d}_i''(t+\Delta t)s\Delta t, \tag{4}$$

where s denotes the speed of each individual (remember, this is a constant). A sample snapshot of the simulation results is shown in Fig. 1.

3 Dimension reduction by diffusion maps

Each "data point" arising during temporal simulations of the particle-based model is a vector in a high-dimensional space (of positions and velocities); the more the particles in the swarm, the higher the dimension of this space. We start with the hypothesis that the behavior of the swarm can be *effectively reduced*; that is, we should be able to write closed causal (deterministic, or possibly Markovian stochastic) evolution equations for a (hopefully small) set of collective swarm descriptors. These descriptors (hopefully far fewer than the dimensionality of the fine-scale particle model) may arise from rigorous mathematical considerations of the model, from intuition/experience with the system, or, - as we describe here- from machine learning (i.e. from mining simulation data); see also the example in [7] and the discussion therein. Principal component analysis (PCA, [8]), a classical and widely used data-based dimensionality reduction technique works best when the data live on or close to a *linear* submanifold (a hyperplane) of the high-dimensional particle model state space. However, PCA is severely limited by its inability to successfully capture *nonlinear dependencies* among the data. Over the past decade or so, nonlinear dimensionality reduction techniques such as local linear embedding (LLE, [9]), Isomap [10], and Laplacian Eigenmaps [2] have been developed to uncover and parametrize low-dimensional *nonlinear manifolds* from high-dimensional datasets, and have attracted great interest. Diffusion maps (DMAP) [4,5] is a relatively recent such nonlinear dimensionality reduction technique, whose more detailed algorithmic description follows below. By constructing a diffusion process across a given, high-dimensional data point set, and by computing the leading spectrum of the associated Markov matrix, this technique has the potential to identify -and, importantly, parameterize- underlying low-dimensional, nonlinear manifolds. Expressing the data points in terms of coordinates *on* this lower dimensional manifold results, therefore, in dimensionality reduction. Figure 2 illustrates this: the given

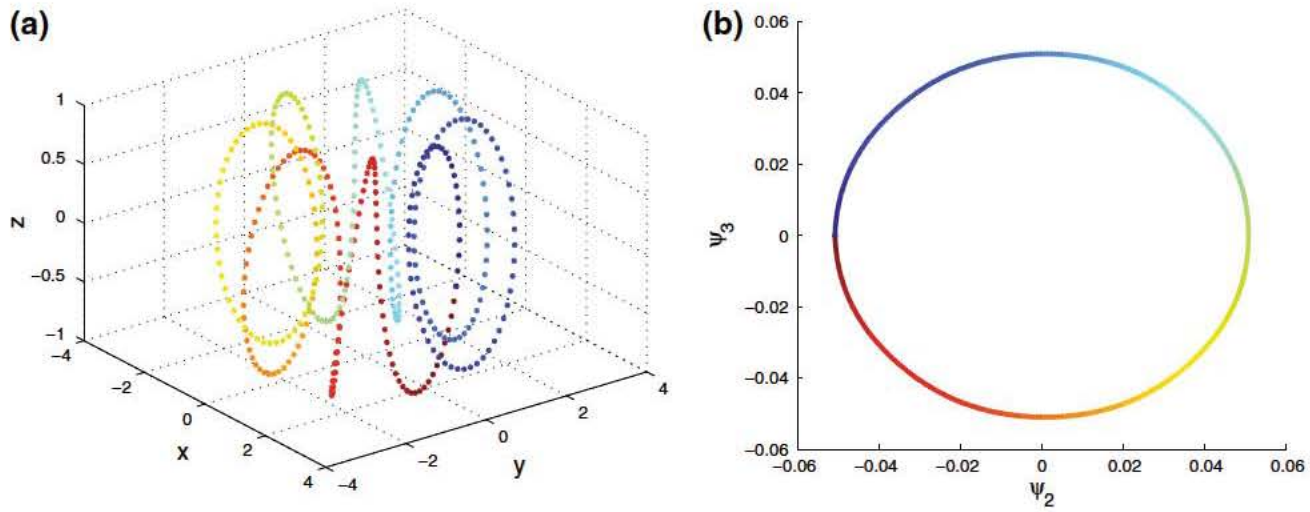

Fig. 2 DMAP illustrative example: **a** The original curve data set in R^3. **b** The DMAP mapping of the closed curve onto a circle in R^2. (ψ_2, ψ_3) are the first two non-trivial DMAP eigenvectors. (Color figure online)

data points, lying on a "slinky" curve, originally live in a three dimensional space; however, the underlying manifold is parameterized by the arc length along the "slinky". By using the first two non-trivial eigenvectors of an appropriately constructed matrix, the DMAP method maps the three-dimensional closed curve onto a circle on the plane, thus reducing the description dimension from three to two.

The DMAP approach is based on the construction of a Markov transition probability matrix corresponding to a random walk on a graph *whose vertices are the data points*. The transition probabilities are computed based on the *local similarities between pairs of data points*. The first few eigenvectors of this Markov matrix are used as coordinates (called the DMAP coordinates); they provide a lower-dimensional (coarser) description of the data points: these are the coarse variables we want to identify, since they parametrize the nonlinear manifold on which the data lives. This manifold is found using only distances between the data; so its geometry and parametrization is *intrinsic* to the dataset, and does not explicitly depend on the original variables in terms of which the fine model is formulated.

DMAP can be considered as a successful generalization of PCA to nonlinear data sets whose intrinsic geometric structure is not known in advance. Different from PCA, the DMAP coordinates provide *nonlinear* reduced embeddings of the data. Another distinct feature is that DMAP uses *local* similarities of *neighboring* data points to infer the global intrinsic geometric structure of the data - our coveted reduced nonlinear embedding. This makes sense because, for data living on a nonlinear curved manifold, a large Euclidean distance in the ambient data space is not necessarily a meaningful measure of "closeness" or "similarity" between pairs of data points; *geodesic distances* along the data (as in Isomap) would be more meaningful. Small Euclidean distances (as compared to the curvature of the manifold) on the other hand would almost always indicate true closeness of the data (see the discussion and figures in [10]).

3.1 The DMAP construction algorithm.

Given a set of N data points $x_1, x_2, \ldots, x_N \in R^d$ (vectors, arising from particle simulation snapshots) the DMAP embedding is constructed as follows [3,4,6]: We first build a weight matrix W using a positive semi-definite kernel function k,

$$W_{ij} = k(d(x_i - x_j)), \tag{5}$$

where d is a relevant distance metric (e.g. it can be the Euclidean distance). A popular choice for k is the Gaussian kernel $k(d(x_i, x_j)) = exp(-d^2(x_i, x_j)/\epsilon)$, although other choices are also possible. The parameter ϵ is a characteristic length scale which corresponds to the bandwidth of the kernel. Note that the matrix elements W_{ij} almost vanish for pairwise distances larger than ϵ, so in practice W is usually a sparse matrix. The matrix W can also be viewed as an adjacency matrix for a graph with n nodes, with W_{ij} being the weight of the edge between nodes i and j. The weight matrix W is then normalized to be row stochastic, by using a diagonal matrix D whose elements are the row sums of W,

$$D_{ii} = \sum_{j=1}^{N} W_{ij}, \tag{6}$$

$$A = D^{-1}W. \tag{7}$$

The matrix A is a Markov transition matrix which describes the transitions of a Markov chain involving the nodes of the graph defined by W. The probability $p^t_{i\to k}$ of a random

walker starting at point i to arrive at point k at time t is given by A^t_{ik}. Then, to compare points i and k at time t, we should compare the rows of A^t,

$$D_t^2(i,j) = \sum_{k=1}^{n} \frac{(p^t_{i\to k} - p^t_{j\to k})^2}{D_{kk}} = \sum_{k=1}^{n} \frac{(A^t_{ik} - A^t_{jk})^2}{D_{kk}}, \tag{8}$$

where D_{kk} is the degree of the vertex k. (This is to make sure each vertex will have a similar contribution to the similarity measure.) We refer to $D_t(i,j)$ as the *diffusion distance* between i and j at time t. It can be related to the eigenvectors and eigenvalues of the Markov transition matrix A as follows:

$$D_t^2(i,j) = \sum_{k=1}^{n} \lambda_k^{2t}(\psi_k(i) - \psi_k(j))^2 \tag{9}$$

where $\psi_1, \psi_2, \ldots, \psi_N$ and $\lambda_1, \lambda_2, \ldots, \lambda_N$ are the eigenvectors and eigenvalues of A respectively. Because A is similar to the symmetric matrix $S = D^{-1/2} W D^{-1/2}$, its eigenvalues $|\lambda_1| \geq |\lambda_1| \geq \cdots \geq |\lambda_N|$ are real. Since A is row stochastic, it can be shown that $\lambda_1 = 1$ and the corresponding eigenvector $\psi_1 = (1, \ldots, 1)^T$. For many practical problems a spectral gap can be observed at some λ_M, e.g., $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_M| \gg |\lambda_{M+1}| \geq \cdots \geq |\lambda_N|$. In such a case the diffusion distance can be approximated as

$$D_t^2(i,j) = \sum_{k=2}^{M} \lambda_k^{2t}(\psi_k(i) - \psi_k(j))^2. \tag{10}$$

The truncated DMAP embedding at time t is then defined as

$$x_i \mapsto (\lambda_2^t \psi_2(i), \lambda_3^t \psi_3(i), \ldots, \lambda_M^t \psi_M(i)). \tag{11}$$

Usually the following $t = 0$ embedding is used,

$$x_i \mapsto (\psi_2(i), \psi_3(i), \ldots, \psi_M(i)). \tag{12}$$

In this paper we call the embedding defined in Eq. (12) *the DMAP embedding*. Basically we are using the components of the data in the first $M-1$ DMAP eigenvectors as our coarse variables, and it is common that $M - 1 \ll d$, where d is the dimension of the space of the given fine scale data before the DMAP technique is applied. The DMAP embedding gives a good low-dimensional representation of the data set *if such a representation exists*. It is also interesting to note that if the data actually do come from a Markovian stochastic process, the eigenvectors and eigenvalues are approximations to the eigenfuctions and eigenvalues of the corresponding backward Fokker–Planck operator [6]. Yet in general, the approach is not used to fit a diffusion to the dynamics that produced the data; it is used to detect (with the help of a possibly "unphysical" diffusion process) a good set of variables that parameterize the (reduced-dimensional) manifold on which the data actually lie. The relation between the eigenfunctions of the Laplacian on a domain and the domain geometry (see for example [11]) is a long-standing harmonic analysis research endeavor.

3.2 Manifold interpolation and the Nyström formula

DMAP can be a useful tool for extracting key descriptors of a data set resulting from a dynamical process. However, to successfully incorporate this tool into a systematic coarse-graining framework, we must be able to find the reduced DMAP coordinates for *new* data points, beyond the ones used to initially identify the useful reduced coordinates. This can be accomplished by manifold interpolation through what is known as the Nyström extension [12]. Before presenting the Nyström formula, we first observe the definition of the DMAP eigenvectors and eigenvalues,

$$\lambda_j \psi_j(i) = \sum_{k=1}^{N} A_{ik}\psi_j(k) = \frac{1}{D_{ii}} \sum_{k=1}^{N} W_{ik}\psi_j(k), \tag{13}$$

Given a new data point $x_{New} \in R^d$, an analogous form of Eq. 13 can be written as

$$\psi_j(new) = \frac{1}{\lambda_j D_{new}} \sum_{k=1}^{N} k(x_{new}, x_k)\psi_j(k), \tag{14}$$

where

$$D_{new} = \sum_{j=1}^{N} k(x_{new}, x_j). \tag{15}$$

Equation 14 is the Nyström formula which we use to find the DMAP coordinates for a new "out of set" data point.

4 DMAP construction and a similarity measure between snapshots

A crucial step in the DMAP construction is the selection of the appropriate pairwise similarity measure between data snapshots. The Euclidean distance is an obvious choice for many data sets; however, often a more "informed" distance may serve better. For example, if the data is characterized by one or more underlying symmetries, these need to be "factored out" before quantifying the distance between data points, and the simple Euclidean distance is not a good choice

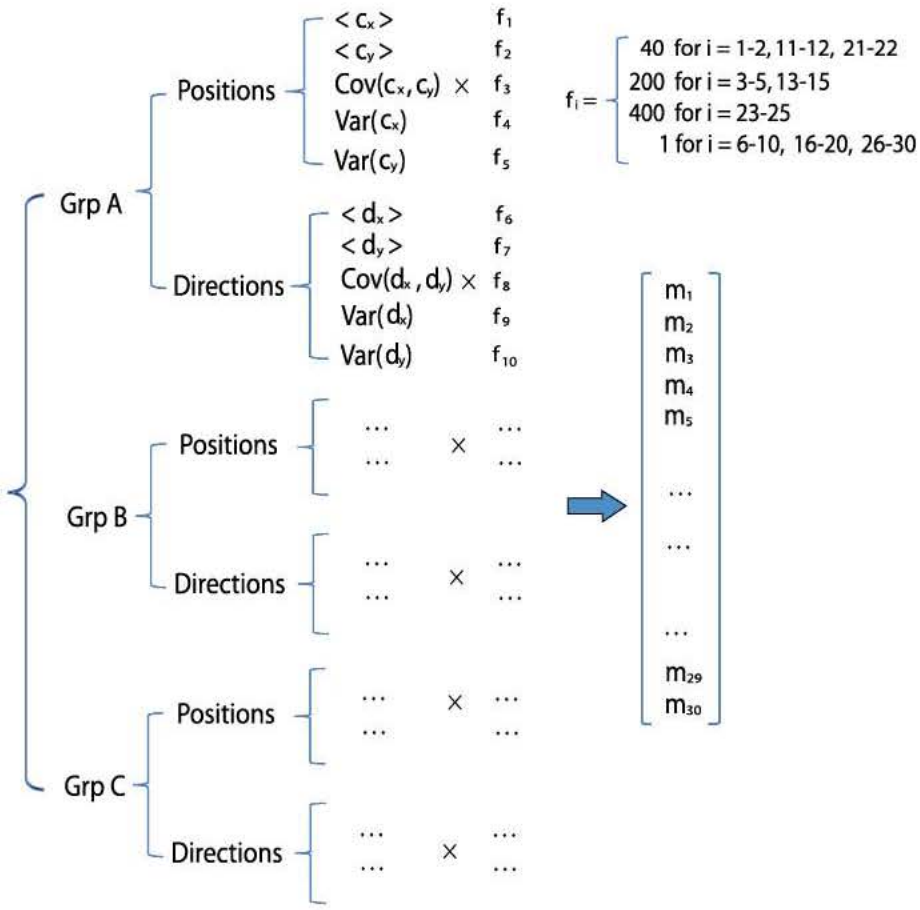

Fig. 3 The ensemble of swarm statistics (intermediate coarse variables) used to formulate our pairwise similarity measure for comparing nearby swarm configurations. There are three different subgroups, each with two properties (x- and y-coordinate related), so in total there are six property groups. For each property group we use the first few statistical moments (mean, variance and covariance in x- and y- related properties) for our representation of the swarm, which gives us a $5 \times 6 = 30$ dimensional "intermediate coarse variable" vector. Each component of this vector is properly normalized, so that all components will have similar magnitude. The weighting factors are chosen based on observations of the simulation results, and the detailed values of these factors are shown in the *top-right* corner of the figure

(see e.g. [13]). For the particle-based animal swarming model we study in this paper, there are three distinct subgroups of agents. Each subgroup is *invariant under a permutation of the identities* of its N_i indistinguishable agents. For this reason, a simple Euclidean distance between the data snapshots is not a useful choice for our problem; it is not clear, when comparing two snapshots, how to order similar particles in each data vector so as to compute a meaningful Euclidean distance.

Optimization-based distance measures, like the Earth Mover's distance [14] are much more appropriate as similarity measures between data snapshots with indistinguishable agents. With this technique, one considers all possible permutations of the particles in each pair of data snapshots and computes all of the corresponding pairwise distances. *The minimum of these distances* is then used as the similarity measure between the two data snapshots. The disadvantage of this approach is the high computational cost, since the enumeration of all possible $n!$ permutations becomes impractical even when n is a modest double digit number. (for recent algorithmic developments in the computation of the Earth-Mover's distance see [15,16].)

In this study, we use instead the first few statistical moments of the particle distributions to construct the similarity measure between each pair of data snapshots (each pair of swarm configurations). Before we describe the details of our similarity measure computation, several terms need to be defined. The microscopic variables of the agent-based animal swarming model are the positions and directions of the individuals in the $2D$ physical space. So for N individuals the microscopic variables have dimension $4N$. We define a set of intermediate coarse variables (swarm statistics, collective swarm descriptors), $\mathbf{x}_i$, which will be used for the computation of the similarity measure $k_{ij} = exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\epsilon}\right)$ between each pair of data points; a *data point* here actually refers to a data snapshot which contains the positions and directions for each of the N agents in the system. These "intermediate variables" $\mathbf{x}_i$ will help identify our "ultimate" coarse variables, the first several DMAP eigenvector components of the data.

The assembly of the intermediate variables x_i is illustrated in the chart shown in Fig. 3. There are three different subgroups, each with two types of properties; so in total there are 6 property-groups. For each property-group we choose to use the first few statistical moments (mean, variance and covariance in x- and y- coordinates) as the coarse representation, which gives us a $5 \times 6 = 30D$ dimensional intermediate variable vector. Before we use it for the similarity computation, each element of this vector is weighted, so that every

element has similar dynamic range. The choice of the weighting factors is made based on observations of the statistics of the simulation results, and the detailed values of these factors are shown in the top-right corner of Fig. 3.

5 A coarse-grained, two-dimensional effective description

Based on the similarity measure chosen in the previous section, we apply DMAP to simulation data from three different cases: (1) a symmetric case, with equal number of individuals in each of the two informed groups ($N_A = 7, N_B = 7, N_U = 36$); (2) an asymmetric case, with slightly more individuals in one of the two informed groups ($N_A = 8, N_B = 7, N_U = 35$); (3) also an asymmetric case, but with lower proportion of informed individuals ($N_A = 6, N_B = 5, N_U = 39$).

To generate the raw data set for the DMAP construction, we run the particle-based simulations and store the data snapshots at a frequency of one snapshot every five time steps. For each case, we generate 60,000 data snapshots, with each data snapshot consisting of positions and directions for all the individuals. However, to best exploit the spatial symmetry of the problem, not all the data snapshots are generated directly from the particle-based simulations. For case (1), because the problem is symmetric about both the x- and the y- axis passing through the center of mass of the swarm, we generated 15,000 data snapshots from the particle-based simulations, and then performed a reflection about x- axis, the y-axis and also about both the x- and y- axis. Here the statement "the problem is symmetric" implies an equivariance of the dynamics: reflecting a simulation snapshot gives the same result as (commutes with) reflecting the initial conditions and evolving them for the same number of steps. However, caution must be used for this case when we perform the reflection about the x- axis: a little thought shows that, in addition to reflecting the positions and the directions of each agent, the identities of the agents of the two informed subgroups also need to be swapped in order to create the appropriate image. For cases (2) and (3), because the problem is only symmetric about the y-axis, we generated 30,000 data snapshots from the particle-based simulations, and then performed the reflection about the y-axis.

Once the raw data set is generated, we compute the intermediate variables defined in the last section—the rescaled first few statistical moments of positions and directions for each subgroup. Following the procedures discussed in Sect. 3, the DMAP embeddings are obtained. Figure 4 shows the DMAP embedding and the associated representative microscopic snapshots for the symmetric case (case (1)). In the embedded two dimensional DMAP space, the first meaningful Diffusion Map eigenvector (ψ_2) corresponds to the up-down direction of collective motion in the original physical space, and the second meaningful DMAP eigenvector (ψ_3) corresponds to the left–right direction of collective motion in the original physical space. For example, for the DMAP embedding located on the middle left of the 2D map ($\psi_2 \sim -5, \psi_3 \sim 0$), the direction of collective motion is vertically upward; for the DMAP embedding located on the bottom of the 2D map ($\psi_2 \sim 0, \psi_3 \sim -0.008$), the direction of collective motion is to the left; at the center of the map ($\psi_2 \sim 0, \psi_3 \sim 0$), the group "jiggles" and there is no consensus among the agents about which way to move. There are also interesting microscopic features for the different collective migration states classified through the DMAP embedding. As Fig. 5a, b show, when the group is in the state of upward collective motion, the group is elongated. The informed agents (red circles) who prefer the upward direction of motion tend to cluster at the moving front of the group to lead the migration. The other subgroup of informed agents (black triangles) who prefer the downward direction of motion tend to spread and stay at the tail of the group. The two subgroups of informed agents appear "separated" by the uninformed ones (blue asterisks).

When, on the other hand, the group is in the state of rightward collective motion, as Fig. 5c, d show, the two subgroups of informed agents tend to spread at the two sides of the bulk of the uninformed agents, with some of the agents from one informed subgroup occupying the moving front of the entire group to lead the migration. In the "jiggling state", the agents from the three subgroups are mixed with each other, and there is no developed consensus yet among the agents about which direction to move.

The DMAP embedding gives us comparable interpretations for the other two cases. Based on the DMAP embeddings computed for all the three cases, we plot the logarithm of the corresponding probability densities in DMAP space. As Fig. 6 shows, in the symmetric case (case 1), the group spends most of its time moving either upward or downward with equal overall probability; in the asymmetric case (case 2), the group spends more time traveling along the preferred direction of one of the informed subgroups: the one which contains more individuals. As the proportion of uninformed the individuals increases (case 3), the group spends less time along the preferred direction of the dominant informed subgroup.

6 The coarse-grained SDE model approximation

In the symmetric case (equal number of informed individuals in each informed group), as Fig. 6a shows, the group spends most of its time in either the state of upward collective motion or downward collective motion. Due to the intrinsic stochastic nature of the motion, detailed simulation shows that the group switches its direction of collective

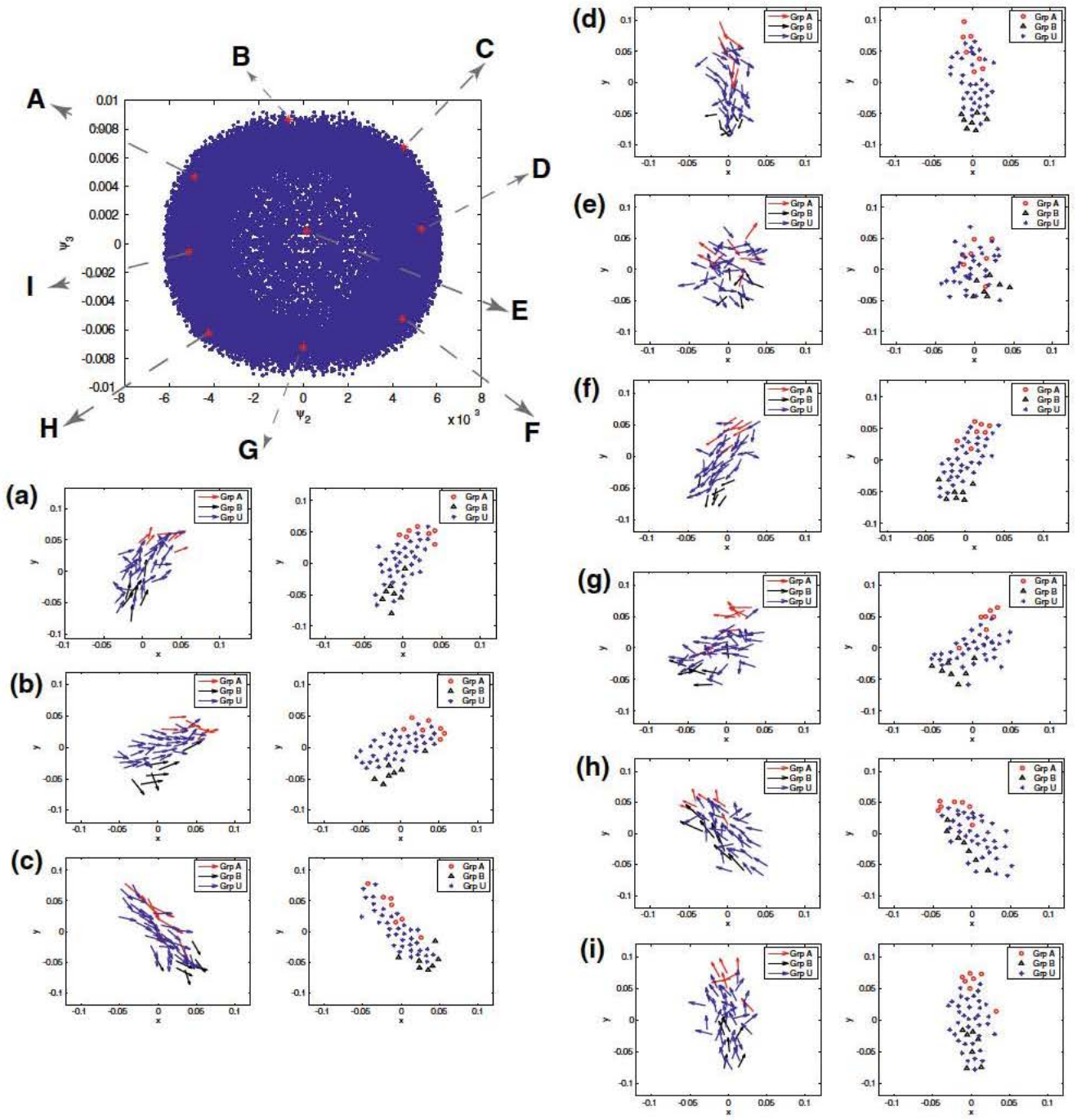

Fig. 4 Diffusion map embedding and the associated representative microscopic snapshots for the symmetric case (case (1)). In the embedded two dimensional Diffusion Map space, the first meaningful diffusion map eigenvector (ψ_2) corresponds to the up-down direction of collective motion in the original physical space, and the second meaningful DMAP eigenvector (ψ_3) corresponds to the left–right collective direction of motion in the original physical space. For each pair of displayed snapshots, the left one presents the direction of motion for the particles; the right one shows the relative positions of the particles. (Color figure online)

motion (from upward to downward and vice versa) at random time intervals. An interesting question naturally arises: how much time (on average) does it take for this switch to occur? In this section, based on the two dimensional DMAP embedding found in the last section, a reduced stochastic differential equation (SDE) model is constructed in order to address this question in a computationally efficient way.

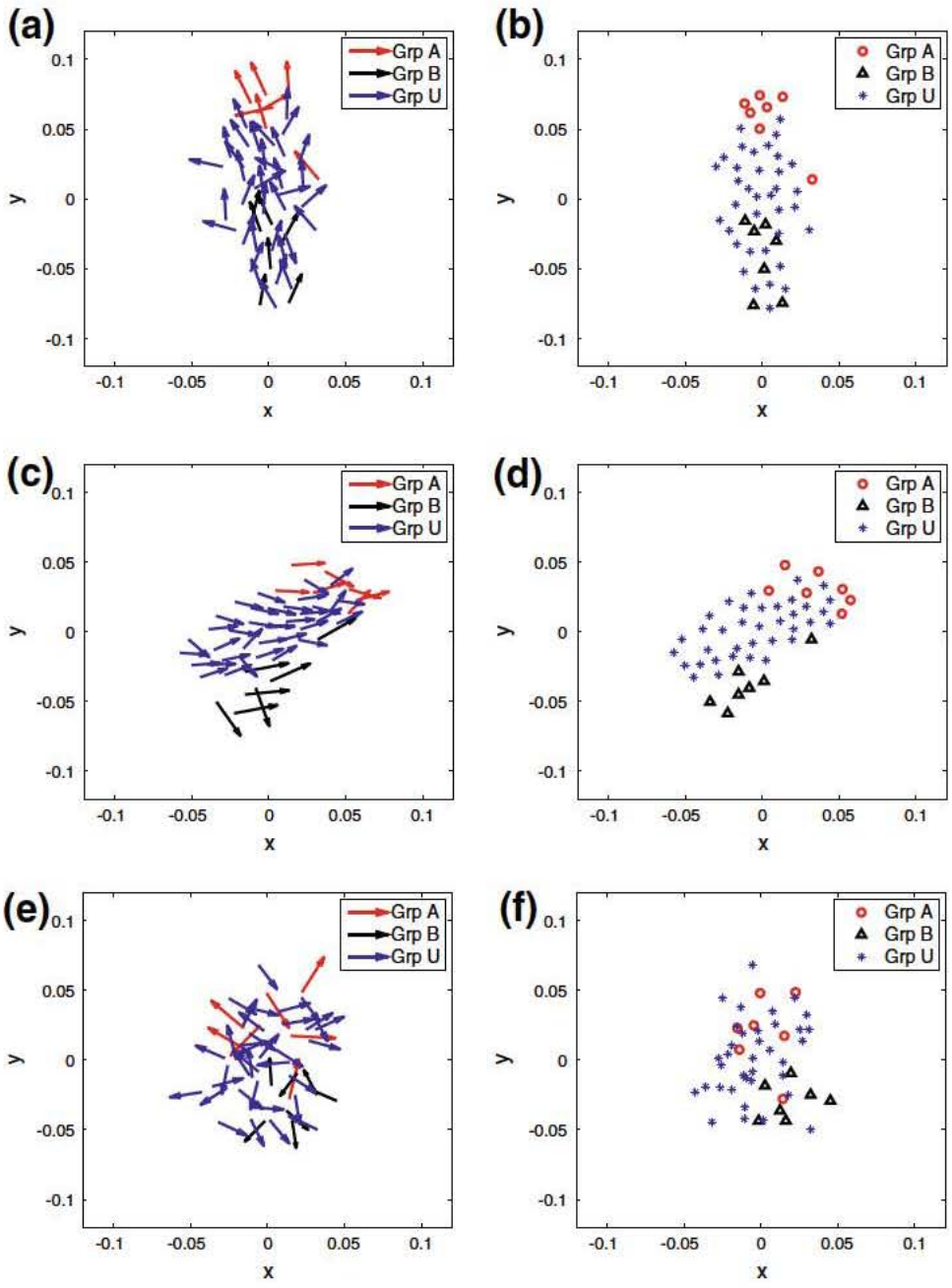

Fig. 5 Sample snapshots of different migration states (case 1). **a**, **b** Upward collective motion. **c**, **d** Rightward collective motion. **e**, **f** "jiggling" state. The *right column* presents the directions of the particles; the *left column* presents the positions of the particles. (Color figure online)

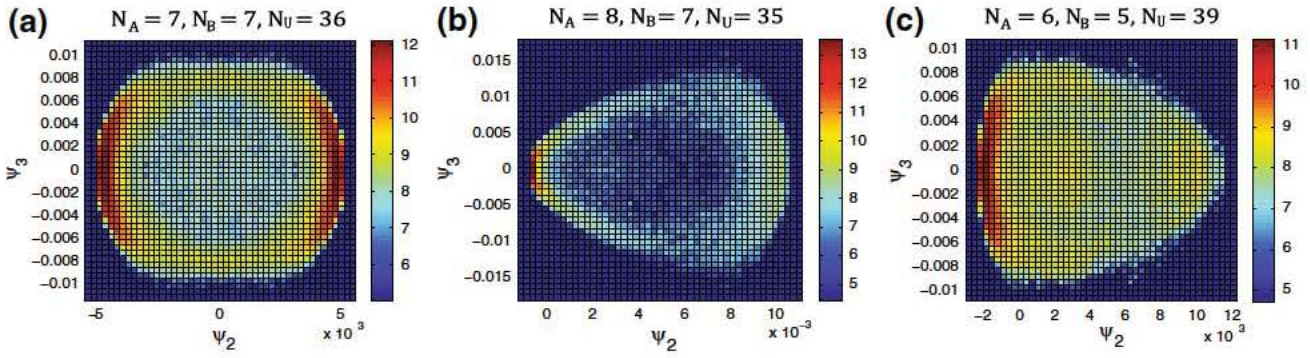

Fig. 6 Plots of log-probability densities for the three cases studied in diffusion map space (the color map is on the right). (Color figure online)

The reduced two dimensional SDE model is in the following form:

$$\begin{aligned} d\psi_2 &= \mu_1 dt + \sigma_1 dW_1 \\ d\psi_3 &= \mu_2 dt + \sigma_{21} dW_1 + \sigma_{22} dW_2 \end{aligned} \tag{16}$$

where W_1 and W_2 denote independent Wiener processes. μ_1, μ_2 are the effective drift coefficients, and $\sigma_1, \sigma_{21}, \sigma_{22}$ are the effective diffusion coefficients. The following equation can be used to estimate these effective coefficients [17]:

$$\begin{aligned} &\mu_i(\psi_2, \psi_3) = \lim_{\Delta t \to 0} \frac{\langle \Psi_i(t+\Delta t) - \Psi_i(t) \rangle}{\Delta t}\Big|_{\Psi_2(t)=\psi_2, \Psi_3(t)=\psi_3} \\ &D_{ij}(\psi_2, \psi_3) \\ &\quad = \lim_{\Delta t \to 0} \frac{\langle (\Psi_i(t+\Delta t) - \Psi_i(t))(\Psi_j(t+\Delta t) - \Psi_j(t)) \rangle}{\Delta t}\Big|_{\Psi_2(t)=\psi_2, \Psi_3(t)=\psi_3}, \end{aligned} \tag{17}$$

where the angled brackets denote expectation and D_{ij} are the diffusion coefficients of the corresponding Fokker–Planck equation for the effective SDE (16). The drift coefficients μ_i govern the deterministic part of the macroscopic dynamics, while the diffusion coefficients D_{ij} represent the stochastic aspect of the SDE. We estimate the drift and diffusion coefficients from finite-length simulations with finite Δt,

$$\begin{aligned} \mu_i(\psi_2, \psi_3) &\approx \frac{\langle \Psi_i(t+\Delta t) - \Psi_i(t) \rangle}{\Delta t}\Big|_{\Psi_2(t)=\psi_2, \Psi_3(t)=\psi_3} \\ D_{ij}(\psi_2, \psi_3) &\approx \frac{\langle (\Psi_i(t+\Delta t) - \Psi_i(t))(\Psi_j(t+\Delta t) - \Psi_j(t)) \rangle}{\Delta t}\Big|_{\Psi_2(t)=\psi_2, \Psi_3(t)=\psi_3} \\ &\quad - \Delta t \mu_i(\psi_2, \psi_3) \mu_j(\psi_2, \psi_3) \end{aligned} \tag{18}$$

where the last term in Eq. (18) helps to correct for the finite size of Δt [18]. The diffusion coefficients in Eq. (16) are estimated via Cholesky decomposition as follows [17],

$$\sigma_1 = \sqrt{D_{11}}, \sigma_{21} = \frac{D_{12}}{\sigma_1}, \sigma_{22} = \sqrt{D_{22} - \sigma_{21}^2}. \tag{19}$$

To numerically estimate the drift and diffusion coefficients using the above formulas, the two-dimensional DMAP space is discretized into 10,000 (100 by 100) small "boxes" or "bins". For each small bin, we locate several instances arising in long-time agent-based simulations, and then for each of these instances, we record DMAP coordinates Δt later. Averaging results as shown above provides numerical estimates of the effective drift and diffusion coefficients. After the drift and diffusion coefficients are estimated, the reduced SDE model is obtained. Figure 7 compares time histories of the agent-based simulations (after converting the solutions to DMAP variables using the Nyström formula discussed in Sect. 3.2) with representative ones from the reduced SDE model. The solutions of the two models appear qualitatively similar to each other.

The important thing to note here is that data mining helped us determine the right dimensionality of the coarse description; this in turn allowed us to "intelligently" sample a much lower-dimensional (just two-dimensional here!) space in order to extract the effective SDE from particle simulation bursts.

7 Mean exit time computation through the coarse-grained model.

Once the reduced SDE model (our effective surrogate model) has been obtained, we are able to efficiently compute various macroscopic level properties such as the mean exit time between dynamically important rare events. In this problem, such interesting events are the switches of the system from "vertical" upward collective motion or downward collective motion to some "non-vertical" state of collective motion. Quantitatively, as Fig. 8 illustrates, we define the mean exit time here as the average time it takes for the system to travel from the red dashed line to the green dashed line. Figure 9 shows the distributions of the exit times for the reduced SDE and the original agent-based model. They qualitatively resemble each other; the statistics of the exit times of the two models are computed and summarized in Table 1, and the results appear reasonably close. Moreover, simulation of the SDE for 1×10^6 steps only takes about half a second, while the original agent-based model requires over 37 min for the same number of time steps (a difference factor of more than 4,000).

8 An open question: predicting the onset of swarm breakup

In the previous section we were reasonably successful in quantitatively answering an important collective dynamics question (distribution of switching times) through a coarse-grained model constructed in terms of the collective variables we obtained through data mining (DMAP). Informed individuals with stronger desire to move toward their preferred direction are less willing to compromise on their preferences for the sake of the group. At high ω, then, there is a significant chance that a group containing informed individuals will separate into independently operating subunits instead of either "jiggling" or traveling as a coherent whole. We thus turn to an open research question: since swarm breakups are often computationally observed, can we successfully computationally predict them as well? Are there coarse-grained variables ("latent" observables, beyond the ones identified above) whose values provide useful predictions of the the swarm breakup? DMAP enables us to use the coarse level dynamics of collective motion to predict the onset of separation events (henceforth known as "breakups") long before they become macroscopically (meaning here, visually) evident to the observer.

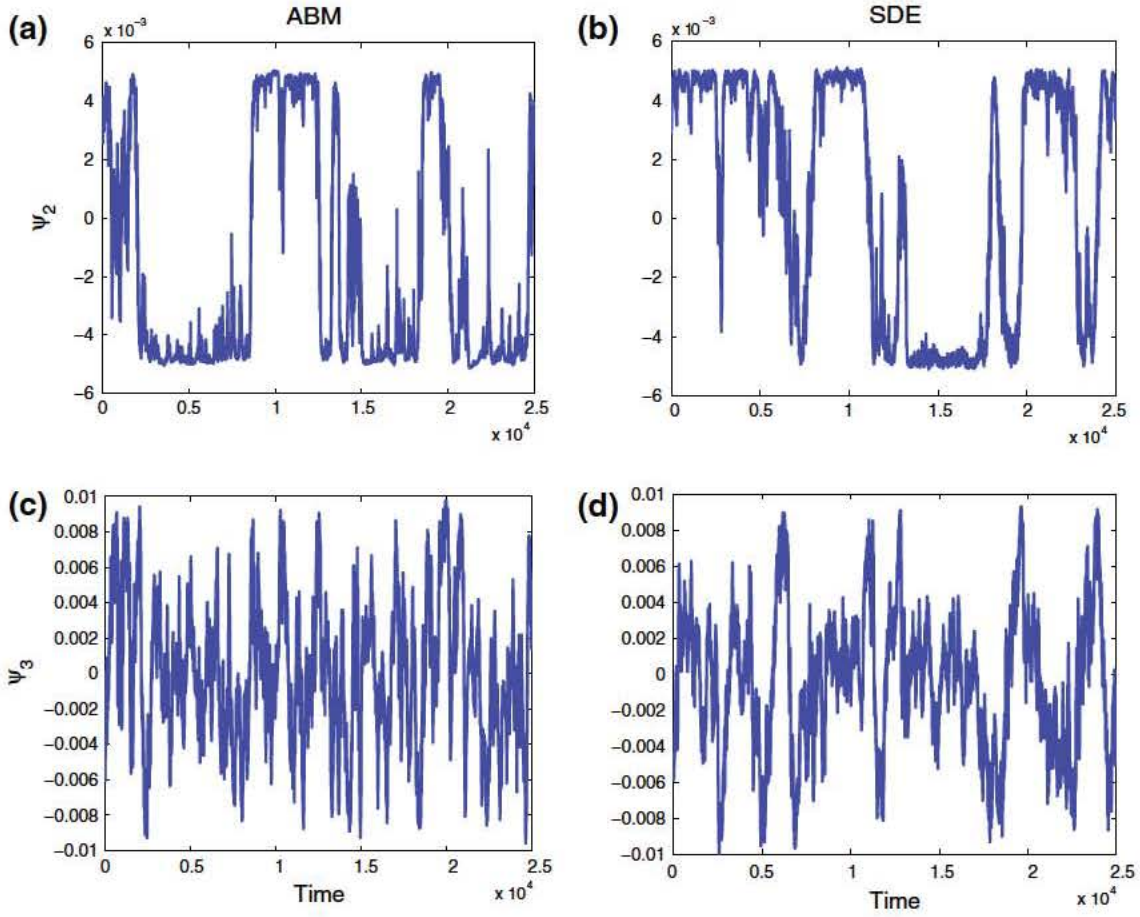

Fig. 7 A comparison between the representative solutions of the agent-based model (**a**, **c**) and the ones from the reduced, effective SDE model (**b**, **d**)

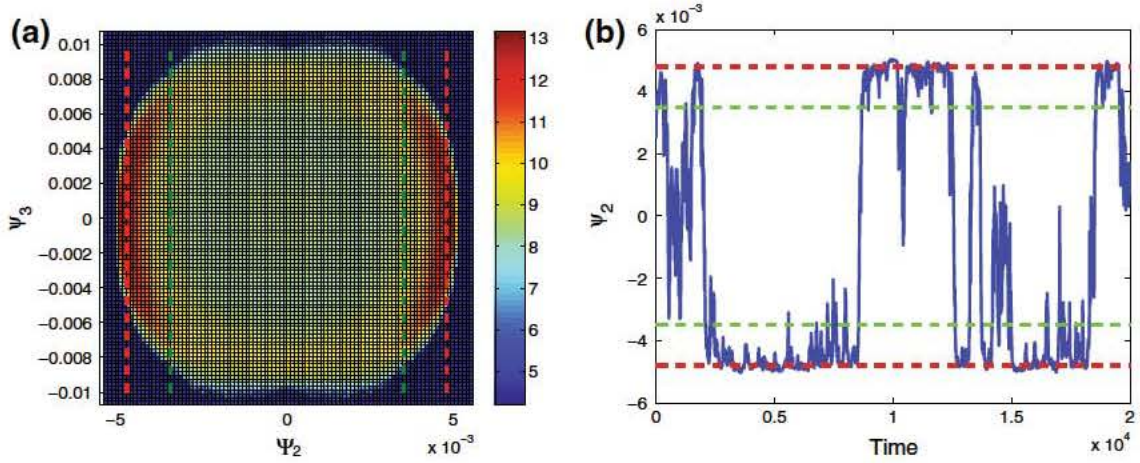

Fig. 8 The mean exit time here is defined as the mean time it takes for the system to travel from the region marked by the *red dashed line* to the region marked by the *green dashed line*. **a** Log-probability density plot for the symmetric (equal informed subpopulation) case; **b** sample time series of the agent-based simulations in the DMAP space. (Color figure online)

In this section, we examine the DMAP embedding for the collective motion simulation when the group is characterized by parameters that generate frequent breakups after a "reasonable" length of simulation time ($N_A = 7$, $N_B = 7$, $N_U = 36$, $\omega = 0.3$). Examining group motion in the period immediately preceding breakup provides some insight into the breakup process. Figure 10 uses representative particle snapshots and the corresponding 2D DMAP embeddings to

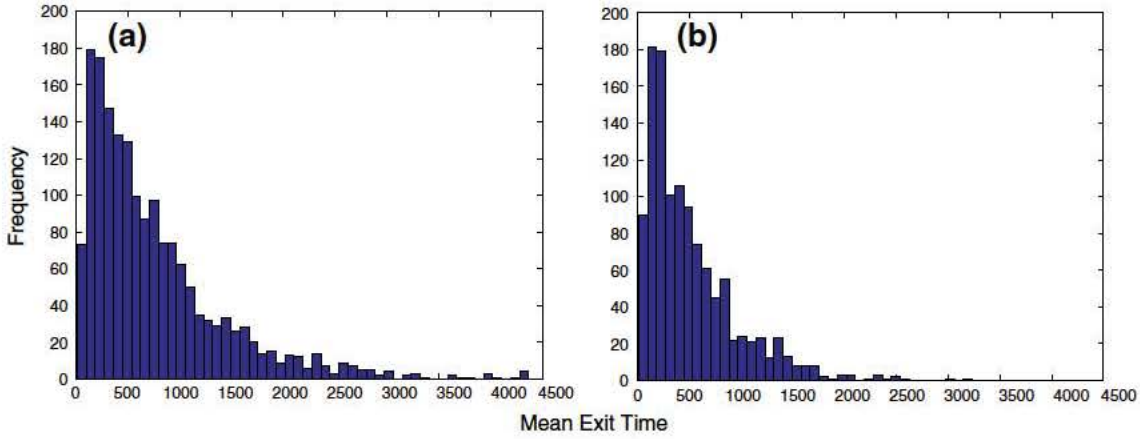

Fig. 9 Comparison of the mean exit time distributions between collective direction switches from **a** the effective reduced SDE and **b** the original particle-based model

Table 1 Comparison of the statistics (the first three moments) of the mean exit time distribution between the original particle-based and the reduced SDE models

Method	Mean	Standard deviation	Skewness
SDE	759.4	677.9	1.874
ABM	511.2	441.2	1.771

illustrate this process for each of the two different breakup types observed. The left column shows the evolution of a "clean" breakup, in which Group A and Group B separate entirely. The right column shows a "complex" breakup, in which one or more members of Group A join the subunit dominated by Group B, or vice versa. The first pair of snapshots in the columns of Fig. 10 show the group prior to the initiation of what we perceive as the breakup process. At this stage, the behavior of the group is consistent with what we have discussed above in Sect. 5. The group moves coherently in the preferred direction of one of the informed subgroups, and the corresponding embedding point is located in the "vertical center" of the two dimensional DMAP space ($\psi_3 \sim 0$). Due to the intrinsic stochastic nature of the motion, however, individuals are not perfectly aligned along the axis of group motion; rather, the normalized individual velocity vectors exhibit a range of divergence from the normalized group velocity vector. If these divergences are large and concentrated enough, individuals may be able to overcome the "pull" of collective motion and turn away from the group, as shown in the second pair of snapshots in Fig. 10. The turning phenomenon is visually apparent in the original physical space, and confirmed by the evident movement of the corresponding DMAP embedding towards the vertical boundaries and horizontal center of the 2D DMAP plane ($\psi_2 \to 0$; $\psi_3 \neq 0$).

If this turning continues, the group will ultimately reach a temporary, apparent "unstable equilibrium" in which the majority of individuals' velocities are oriented perpendicularly to the original direction of motion (in between the preferred directions of the two informed subgroups). This apparent unstable state may be resolved in one of three ways. Collective turning may continue or reverse itself, resulting in a return to coherent motion along the preferred direction of one of the informed subgroups. Alternatively, informed individuals may turn towards their respectively preferred directions, causing a group breakup as the informed individuals lead uninformed followers in opposite directions. We can further classify such breakups depending on how successful informed individuals are at assuming their preferred directions. In "clean" breakups, Group A individuals are clustered close enough to one another, and far enough from Group B individuals during the unstable stage, that the two informed groups separate into entirely separate subunits (Fig. 10c). In "complex" breakups, one or more Group A individuals end up close enough to the Group B cluster, and far enough from the Group A cluster, that the individual(s) must remain with Group B in order to avoid being left out of a subunit altogether, and vice versa (Fig. 10f).

The 2D DMAP embedding helps us to better visualize the mechanism by which changes in individual orientation may lead to a group breakup: although the group can stay together even when many individuals deviate somewhat from group direction, coherence is "threatened" when the sharp divergence of a few individuals triggers mass turning. Additional DMAP eigenvectors may help identify factors that cause individuals to begin to turn away from the rest of the group in the first place.

The fourth-most dominant eigenvector (ψ_5) is an especially promising candidate for such a "latent breakup predictor". Figure 11 shows the same series of microscopic snapshots contained in Fig. 10 along with the corresponding DMAP embeddings based on ψ_2 and ψ_5. There is a clear relationship between ψ_5 and proximity to breakup: ψ_5

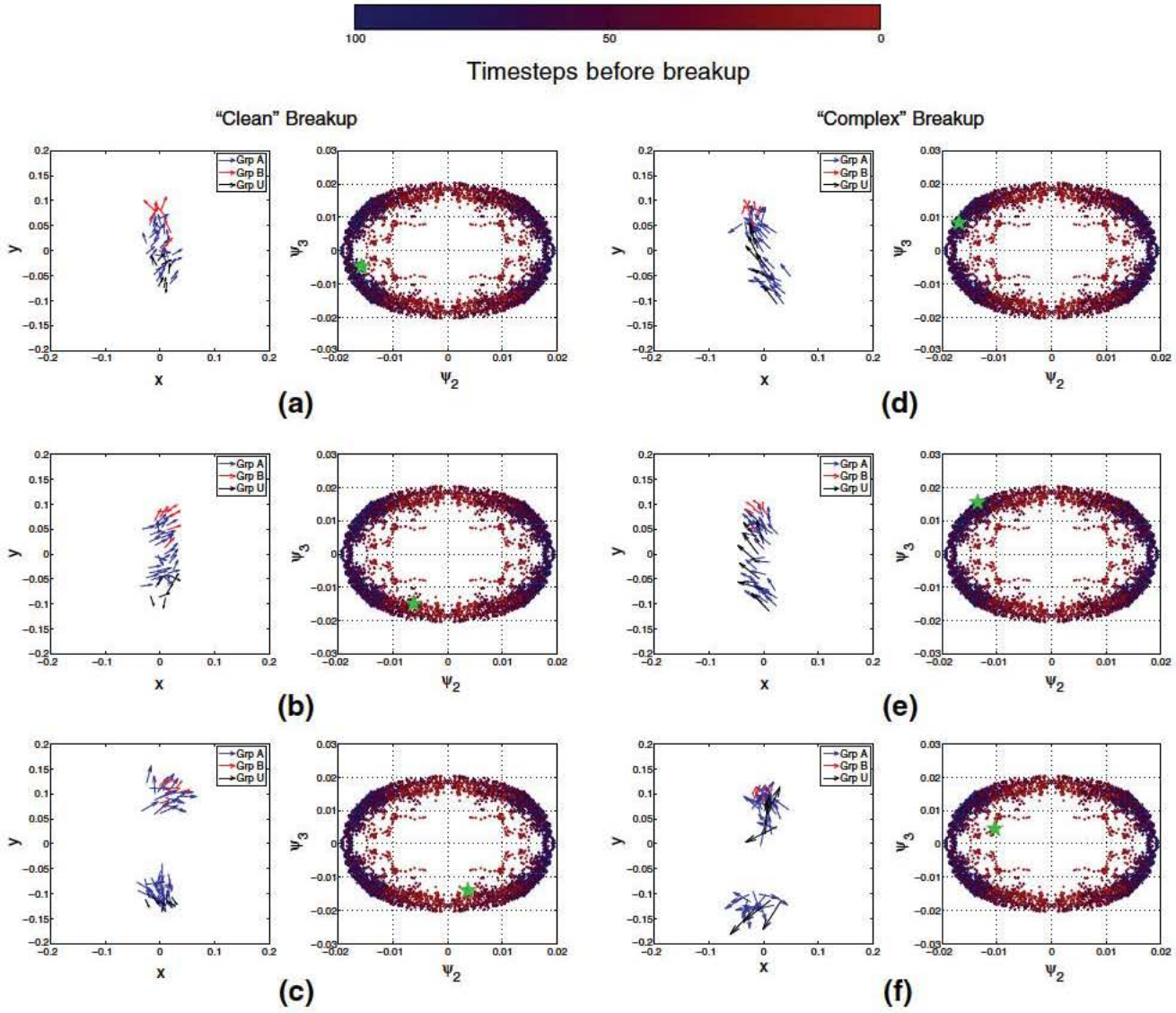

Fig. 10 Sample snapshots and associated 2D DMAP embeddings for different stages of the breakup process for "clean" and "complex" breakups. In each subfigure, the embedded location of the snapshot is marked by a *green star*. **a**, **b** and **d**, **e** In both cases, the group transitions from coherent motion along the preferred direction of one of the informed subgroups to an apparent "unstable equilibrium" in which most individuals are oriented perpendicularly to the original direction of travel. **c** In a "clean" breakup, this apparent unstable equilibrium is resolved by the separation of the group into two subunits, each consisting of uninformed followers led by either Group A or Group B informed individuals. No members of Group A are present in the Group B subunit. **f** In a "complex" breakup, at least one member of Group A is present in the Group B subunit, or vice versa. (Color figure online)

increases (going from negative to positive) as the breakup point approaches. However, it is not immediately apparent which of the (easily physically interpretable) intermediate variables used to construct the DMAP embedding is "mainly" responsible for this relationship. This illustrates a fundamental issue that arises in using data-mining-based variables. While such variables will provide useful parsimonious embeddings, and can lead to dramatically reduced (and thus computationally convenient) dynamic models, the variables have no obvious physical meaning. The same can also be said for principal components: linear combinations of state variables have no obvious physical interpretation / do not necessarily help understanding mechanisms. This should not be surprising—since DMAP only "see" scalar distances between data points, and have no direct information about the original, high dimensional state space, one should not expect them to be easily cast in terms of the original state space coordinates. It is the modeler's task to posit, and test, whether interesting/informative physical variables may be one-to-one, over the data set, with the DMAP variables; so that even though they are not the same, they both can be used to parametrize the manifold on which the data lie. Two use-

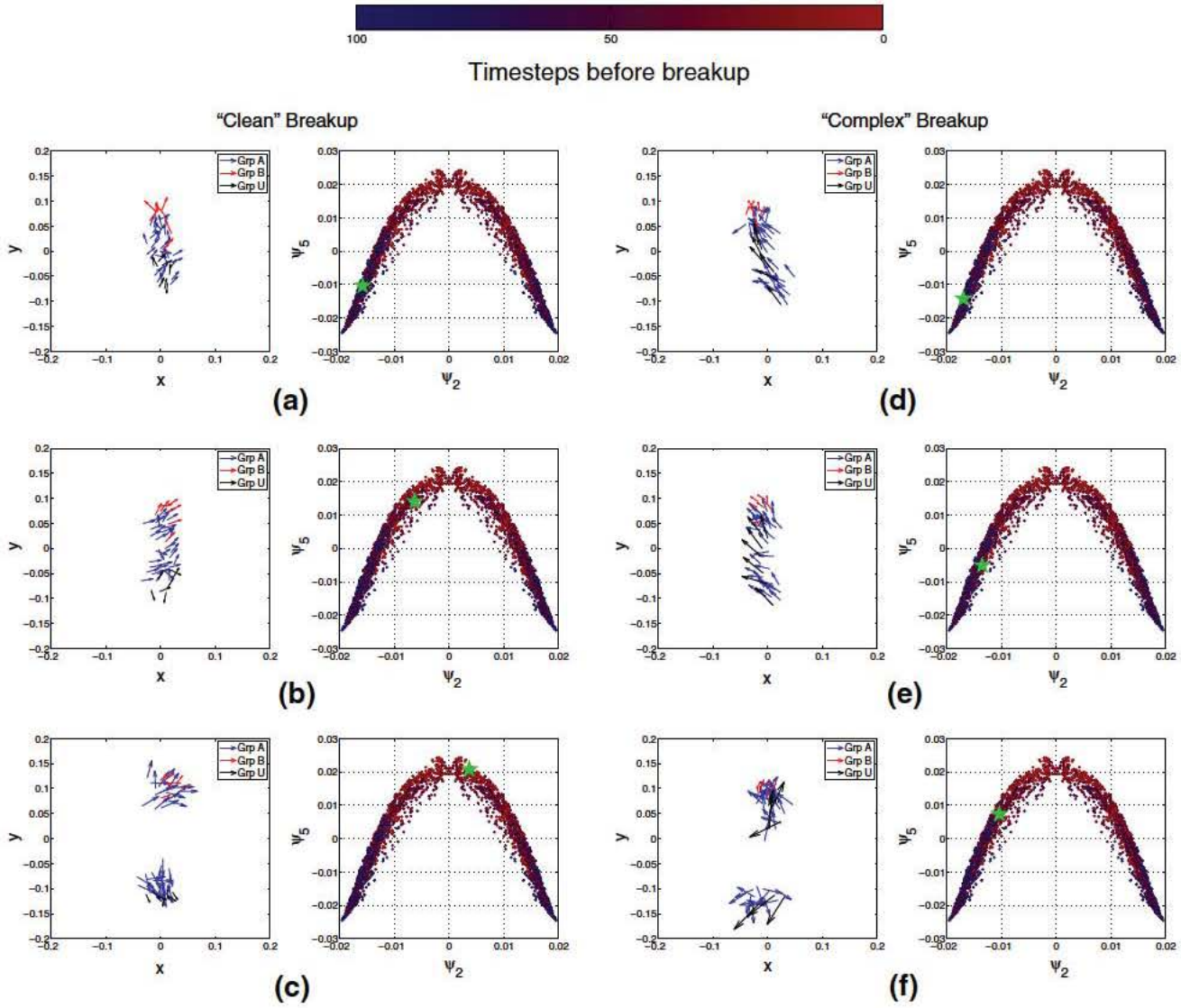

Fig. 11 The DMAP embedding based on ψ_2 and ψ_5 reveals a significant positive relationship between ψ_5 and proximity to breakup. (Color figure online)

ful examples of such a "post mortem" search for physical interpretation of DMAP variables can be found in [7,19].

9 Conclusion

In most contemporary particle based computations (from DEM simulations of multiphase flows (e.g. [20]) to the mechanistic modeling of swarms (e.g. [21]) there is a compelling need for model reduction; and lacking theory-driven or physical-experience driven selections of "the right" collective particle descriptors, it is natural to turn to data-driven, machine learning computational approaches to selecting such variables by mining detailed simulation results. Here we illustrated the use of nonlinear dimensionality reduction techniques, and in particular diffusion maps (DMAP), to systematically extract an effective macroscopic description for a complex particle-based animal swarming model. The advantage of this approach is that it sidesteps the development of extensive physical intuition about the problem, and/or the great difficulty of direct mathematics-based model reduction.

The first few statistical moments of the properties of each particle subgroup were used as a set of "intermediate variables" that helped define an informative pairwise similarity measure between data snapshots. Using this similarity measure, the constructed low dimensional ($2D$) DMAP embedding efficiently characterized and quantified the collective up-down and left–right directions of motion for the group of swarming particles. Based on the identified DMAP coarse variables, we also showed how to construct an effective

reduced model, here in the form of a two dimensional stochastic different equation (SDE); use of this surrogate model can greatly accelerate the computer-assisted approximation of features of the collective dynamics. Our particular illustration consisted of approximating the "switching time" distribution between the two preferred directions of the swarm. The sampled time series of the reduced SDE solution, as well as the statistics of the exit time distribution indeed closely resemble those obtained from the original agent-based simulations with significantly higher computational cost. We also explored the appearance (in DMAP space) of swarm breakup events, and identified a "latent" DMAP variable that stronly correlates with these events; this constitutes a good candidate breakup predictor.

As a preliminary study, it is promising that the DMAP variables identified in this work successfully captured macroscopic level features of the collective direction of motion for the group of simulated agents. In some particular cases, these coarse features might also be "deducible" by human thought (i.e. we can roughly tell the collective direction of motion by looking at a given data snapshot after observing a few simulations). To make our approach more appealing, it is important for future research to explore more "difficult", less intuitive cases, whose coarse features are not at easily perceived by a human - this was exemplified by the study of how a coherent group of migrating animals might *break into separate groups*, and the coarse level dynamics associated with the fragmentation process.

We close by noting the natural link that exists between this data mining approach and the so-called *equation-free* multiscale computation framework [22,23] developed in our group over the years. The idea is to circumvent the (often impractical) derivation of macroscopic closed effective equations for complex (in our case, interacting particle) systems; instead, by using short bursts of appropriately initialized fine scale simulation, and processing the results of these short bursts, we can in effect *solve* the unavailable macro-equations without ever deriving them in closed form. The technique involves frequent "translations" between fine scale and coarse scale descriptions ("lifting" from coarse to fine and "restriction" from fine to coarse states respectively). But in order to do all that, we at least need to know *what the right collective variables* are. In what we presented here, we go beyond equation-free, to "equation- and variable-free" computation: both the relevant variables *and* the information required to solve the unavailable evolution equations for these variables, come from processing short bursts of simulation results (e.g. see [24,25]).

In the current study, the lifting step of the equation-free approach (the construction of fine scale particle swarm realizations consistent with given coarse DMAP variable values) was implemented by running long time particle-based simulations and then creating an "atlas" of eligible fine-scale realizations for a discretization of two-dimensional DMAP space. The drawback of this approach is that some (rarely visited) coarse value bins may contain insufficient data points for the ensemble average computations. It is therefore important for future research to design more advanced lifting procedures, so that microscopic configurations consistent with prescribed macro-variable values can be constructed systematically and economically. We believe that, as more effort is invested in these directions, and more experienced is acquired, this integration of modern data-mining with multiscale mathematics and numerics will become a "standard" tool for the extraction of system-level, coarse-grained information from particle-level dynamics; and that it has the potential to drastically reduce the associated, often prohibitive, computational cost.

Acknowledgments This work was partially supported by the US AFOSR and the National Science Foundation.

References

1. Couzin ID, Krause J, Franks NR, Levin SA (2005) Effective leadership and decision-making in animal groups on the move. Nature 433(7025):513–516
2. Belkin M, Niyogi P (2003) Laplacian eigenmaps for dimensionality reduction and data representation. Neural Comput 15(6): 1373–1396
3. Coifman R, Lafon S, Lee A, Maggioni M, Nadler B, Warner F, Zucker S (2005) Geometric diffusions as a tool for harmonic analysis and structure definition of data: diffusion maps. PNAS 102:7426C7431
4. Coifman R, Lafon S, Lee A, Maggioni M, Nadler B, Warner F, Zucker S (2005) Geometric diffusions as a tool for harmonic analysis and structure definition of data: multiscale methods. PNAS 102:7432–7437
5. Coifman RR, Lafon S (2006) Diffusion maps. Appl Comput Harmon Anal 21(1):5–30
6. Nadler B, Lafon S, Coifman RR, Kevrekidis IG (2006) Diffusion maps, spectral clustering and reaction coordinates of dynamical systems. Appl Comput Harmon Anal 21:113–127
7. Frewen TA, Couzin ID, Kolpas A, Moehlis J, Coifman RR, Kevrekidis IG (2010) Coarse collective dynamics of animal groups. In: Gorban AN, Roose D (eds) Coping with complexity: model reduction and data analysis. Springer Lecture Notes in Computational Science and Engineering 75:299–310
8. Jolliffe IT (2002) Principal component analysis. Springer, New York
9. Roweis ST, Saul LK (2000) Nonlinear dimensionality reduction by locally linear embedding. Science 290(5500):2323
10. Tenenbaum JB, de Silva V, Langford JC (2000) A global geometric framework for nonlinear dimensionality reduction. Science 290(5500):2319
11. Kac M (1966) Can you hear the shape of the drum? Am Math Mon 73(4):1–23
12. Bengio Y, Delalleau O, Roux N, Paiement JF, Vincent P, Ouimet M (2004) Learning eigenfunctions links spectral embedding and kernel PCA. Neural Comput 16(10):2197–2219
13. Sonday BE, Singer A, Kevrekidis IG (2013) Noisy dynamic simulations in the presence of symmetry: data alignment and model reduction. CAMWA 65(10):1535–1557

14. Rubner Y, Tomasi C, Guibas LJ (2000) The earth movers distance as a metric for image retrieval. Int J Comput Vis 40(2):99–121
15. Coifman RR, Leeb WE (2013) Earth mover's distance and equivalent metrics for spaces with semigroups. Yale University Computer Science Technical Reports, YALEU/DCS/TR1481, July 22
16. Coifman RR, Leeb WE (2013) Earth mover's distance and equivalent metrics for spaces with hierarchical partition trees. Yale University Computer Science Technical Reports, YALEU/DCS/TR1482, July 22
17. Laing CR, Frewen T, Kevrekidis IG (2010) Reduced models for binocular rivalry. J Comput Neurosci 28(3):459–476
18. Ragwitz M, Kantz H (2001) Indispensable finite time corrections for Fokker–Planck equations from time series data. Phys Rev Lett 87(25):254501
19. Sonday B, Haataja M, Kevrekidis IG (2009) Coarse-graining the dynamics of a driven interface in the presence of mobile impurities: effective description via diffusion maps. Phys Rev E 80:031102
20. Moon SJ, Sundaresan S, Kevrekidis IG (2007) Coarse-grained computations of demixing in dense gas-fluidized beds. Phys Rev E 75:051309
21. Zohdi TI (2009) Mechanistic modeling of swarms. CMAME 198:2039–2051
22. Kevrekidis IG, Gear CW, Hummer G (2004) Equation-free: the computer-aided analysis of complex multiscale systems. AIChE J 50(7):1346–1355
23. Kevrekidis IG, William Gear C, Hyman JM, Kevrekidis PG, Runborg O, Theodoropoulos C (2003) Equation-free coarse-grained multiscale computation: enabling microscopic simulators to perform system-level tasks. Comm Math Sci 1(4):715–762
24. Erban R, Frewen TA, Wang X, Elston TC, Coifman R, Nadler B, Kevrekidis IG (2007) Variable-free exploration of stochastic models: a gene regulatory network example. J Chem Phys 126:155103
25. Laing CR, Frewen T, Kevrekidis IG (2010) Reduced models for binocular rivalry. J Comp Neurosci 28:459–476