*Proceedings of the American Control Conference*, Montreal, Canada, 2012

# In the dance studio: Analysis of human flocking

Naomi E. Leonard, George Young, Kelsey Hochgraf, Daniel Swain, Aaron Trippe, Willa Chen, Susan Marshall

***Abstract*— Flock Logic is an art and engineering project that explores how the feedback laws used to model flocking translate when applied by a group of dancers. The artistic goal is to create tools for choreography by leveraging dynamics of multi-agent systems with designed feedback and interaction. The engineering goal is to develop insights and design principles for multi-agent systems, such as human crowds, animal groups and mobile robotic networks, by examining the connections between what individual dancers do and what emerges at the level of the group. We describe our methods to create dance and investigate collective motion. To illustrate, we analyze the overhead video of an experiment in which thirteen dancers moved according to simple rules of cohesion and repulsion in response to the relative position and motion of their neighbors. Importantly, because we have prescribed the interaction protocol, we can estimate from the tracked trajectories the time-varying graph that defines who is responding to whom as time evolves. We compute time-varying status of nodes in the graph and infer conditions under which certain individuals emerge as leaders.**

## I. INTRODUCTION

The *Flock Logic* project [1] is inspired by the complex and beautiful motion of bird flocks and fish schools and, in particular, by the understanding that collective animal group motion emerges not from a prescribed choreography nor even from a designated leader, but rather from simple rules of response that each individual obeys. These feedback rules govern how each individual moves in response to the relative position or motion of its close neighbors. For instance, basic flocking rules typically have a cohesive element and a repulsive element [2]. The cohesive element requires that while each individual moves around it should remain a comfortable distance from a few others; the repulsive element requires that each individual should move away from others that get too close in order to avoid collision.

Flock Logic explores what happens when a group of dancers apply these and related feedback laws as they move around a space together. If the prescribed feedback rules approximate well how individual birds and fish interact, and if the dancers carry out the feedback laws faithfully, then the motion of the group resembles flocking or schooling. However, unlike in animal groups where the individual rules and interactions are only hypothesized, in the human groups where the individual rules and interactions are designed, we can use a systematic approach to understand how individual-level behaviors connect to the aesthetics and functionality of the emergent group-level behaviors.

Supported in part by the Essig Enright Fund, Princeton University, NSF grant ECCS-1135724, AFOSR grant FA9550-07-1-0-0528 and ONR grant N00014-09-1-1074. N.E. Leonard, G. Young, K. Hochgraf and D. Swain are with the Dept. of Mechanical and Aerospace Engineering, A. Trippe and W. Chen are with the Dept. of Computer Science and S. Marshall is with the Program in Dance and Lewis Center for the Arts, Princeton University, Princeton, NJ 08544, USA. naomi@princeton.edu

Heterogeneity in the group enters naturally as people may respond differently to different people and may prioritize rules and resolve conflicts differently. This affects how information passes through the group and how the group as a whole responds to external forces. By further imposing different feedback and interconnection rules, environmental signals and perturbations, and heterogeneity in information, we create a wide range of artistic and scientific possibilities.

Our artistic goal is to develop the means to enable interesting, appealing collective motion that emerges from individual interactions; we have focused on the influence of information passing, local versus global sensing, relative motion, heterogeneity, perturbations, and external pressures. In theater and dance, there is a long history of performance derived from rules and games [3]. Contemporary choreographer Forsythe has studied synchrony and pattern in dance [4]. Sgorbati explores dance through “Emergent Improvisation,” which is modeled after ordering principles observed in nature [5].

Our engineering goal is to gain insight into mechanisms of animal group and human crowd dynamics and into design principles for control of natural and robotic groups. Dancers moving in a studio, responding to local neighbors and environment, provide a reasonable approximation to the collective motion of a mammal herd. The walls of the studio are like trees or topography, and the heterogeneity among the dancers (experience, height, confidence) is similar to that in a herd [6]. Dancers are particularly well suited because they are trained to be physically aware and can comfortably handle a number of feedback rules. We aim to understand from the human data how influence among individuals in the network is distributed and how that is reflected in the spatial distribution of individuals and in the group-level shape and motion dynamics. This could, e.g., lead to insights on how animals organize themselves to reduce vulnerability to predators [7], and to bio-inspired methods for designing robust and responsive networks of heterogeneous robots.

Motivating and complementary scientific studies of human collective motion, many of which focus on crowd dynamics, do not, as far as we are aware, engage dancers as participants. Experiments on leadership and decision making in human crowds are described in [8]. In [9] analysis of natural pedestrian group motion revealed the influence of social interactions on crowd dynamics. In [10] a design method for human collective behaviors used evolutionary dynamics.

In this paper, we describe the Flock Logic project and tools used for our artistic and engineering investigations. As an illustration, we examine one experiment with thirteen dancers in a studio following the flocking rules of cohesion and repulsion. Using the trajectories tracked from an overhead video camera and the prescribed interaction rules, we estimate the time-varying graph that encodes who is sensing whom as a function of time. We compute the time-varying status of each node in the graph and use these to infer emergent leaders.

In Section II we describe the project, our on-line FlockMaker software tool, and the human flocking experiments. Trajectory tracking is described in Section III. In Section IV we review graphs and FlockGrapher, our tool for visualizing graphs. In Section V, we estimate the time-varying graph of the network. In Section VI we estimate node status and discuss the influence of individuals. We conclude in Section VII.

## II. HUMAN FLOCKING

### A. Flock Logic

Flock Logic began as a collaboration between a control engineer (Leonard) and a choreographer (Marshall) to explore artistically and scientifically how individual rules of interaction and response within a network of dancers yield complex emergent collective motion of the group. Initially the exploration was with professional dancers and then subsequently with students, most of whom had previous dance training, as part of a semester-long course in Fall 2010.

To generate human flocking, the dancers were asked to move around a space and follow rules that were defined in advance. To enable *cohesion*, each dancer was given the rule to keep $m$ of their neighbors at a distance of arm's length with the selection of the $m$ neighbors freely changeable. To enable *repulsion*, each dancer was asked to avoid letting any dancer get closer than arm's length. To prevent tripping, the dancers were asked to avoid moving backwards.

These three rules (cohesion, repulsion, backwards avoidance) were among the most fundamental rules examined, and yet rich and beautiful collective behaviors were routinely observed. In part because each dancer's motion was relatively under-prescribed, there was considerable room for variation among individuals, e.g., in speed, facing direction relative to motion, selection of neighbors, positioning relative to neighbors, and response to walls or obstacles.

Variations on the three fundamental rules were prescribed as well as a range of additional and alternative rules. For example, rules for alignment with neighbors, response to obstacles and walls, motion in the vertical, arm movements, etc., were implemented. More complex informational structures were imposed – for example, two or three dancers in the group were secretly given additional rules, such as to move to a particular location or according to a particular pattern. The dancers also performed rules for other kinds of behaviors such as dynamic coverage and pursuit and evasion.

In early December 2010, volunteers from the community joined the dancers in two site-specific performance events. Video clips from these events are publicly available [1].

### B. FlockMaker

*FlockMaker* is a Java WebStart application developed to aid the Flock Logic project and designed for simulation and exploration of collective motion [11]. Although intuitive for a curious layperson, FlockMaker has the capability to model complex combinations of flocking rules and initial configurations. In the model, each dancer is represented as a single particle in two-dimensional space, with variable velocity. Speed and facing angle (but not acceleration) are taken to be approximately continuous in time.

The simulation user can assign a variety of flocking rules to the dancers, such as "Pursue Someone," "Repel Neighbors," and "Slow Down Near Neighbors." To further control behavior, the user can set values for a wide range of parameters pertaining to a dancer's rules or initial configuration, including radius of sensing, number of neighbors sensed, maximum speed of rotation, and magnitude of random noise. Different rules can be assigned to different dancers. Furthermore, each dancer can be assigned to follow multiple rules at a time, each rule potentially carrying a different relative weight representing its level of priority.

Dancers interact not only with each other, but also with the room in which they are moving, represented as a rectangular space contained within four walls. The FlockMaker user can change the size of the room, add obstacles to the room, and add rules applicable only within certain zones of the room.

### C. Experiments

A series of human flocking experiments was run in mid December 2010 in the 62' 7" x 28' 4" New South dance studio at Princeton University. Groups of dancers carried out the three basic rules of flocking with manipulations on initial conditions, number of dancers $N$, and number of neighbors $m$ for cohesion. Alignment with neighbors was tested as was the assignment of an additional rule to two of the dancers (of which the others were not aware). The dancers also implemented the rules for cyclic pursuit.

Six Trendnet IP-600 cameras, synchronized over a local wired network, were set up in fixed locations to record the motion of the dancers. Two cameras were hung on the ceiling near either end of the studio, facing inward towards each other, and four were mounted high up on one side wall. Camera views covered the majority of the space in the studio and overlapped significantly. Using built-in software, the cameras recorded video and stored it on a laptop. The video provided 640 x 480 resolution and 20 frames per second.

For the December 2010 series of experiments, part of the room was blocked off so that the motion of all of the dancers could be fully captured by one of the six cameras (one of the two fixed to the ceiling). The dancers wore bright colored hats, black clothing and bare feet to aid trajectory tracking.

In this paper we examine one experiment from the series in which there were $N = 13$ dancers – two professional dancers and eleven students. All thirteen dancers were asked only to follow the three basic rules of flocking with cohesion to $m = 2$ neighbors. The total time for the experiment was 185 seconds, corresponding to the period from the start to the stop of the music. We study the tracked trajectories of the dancers from the first 72 seconds of this experiment.

## III. TRAJECTORY TRACKING

Trajectories were estimated using custom tracking software applied to the overhead video from one camera for the first 72 seconds of the experiment. The tracked trajectories comprise an ordered set of 1440 planar position vectors $(x, y)$ for each of the thirteen dancers. A velocity vector is computed for each dancer at every time step by differencing the position vectors. Speed and heading are computed as the magnitude and angle of the velocity vector. Figure 1 shows one frame from the video with superimposed tracked positions and directions of motion.

The custom tracking software uses a modified version of a real-time tracking algorithm that we have developed and used successfully for experiments involving multiple fish and robots [12]. The algorithm is implemented using the MADTraC C++ library [13], which in turn relies upon OpenCV [14] for low-level image processing routines. The original tracking software was designed to address the challenges of tracking potentially densely distributed objects that are very similar to one another in appearance. It was therefore applicable to the task of estimating dancers' trajectories.

The tracking algorithm follows three steps that are iterated for each video frame. In the first step, image segmentation produces a set of "blobs", such that each blob is a collection of contiguous pixels with high likelihood of belonging to any dancer's hat. Likelihood is determined by thresholding each pixel's value in HSV color space and mapping to a binary image. Blobs are extracted from the binary image using OpenCV's built-in blob labelling algorithm, which is based on [15]. A blob is often associated with more than one dancer because of the physical proximity of dancers to one another, the proximity of dancers in the image due to the angle of the camera and noise in the image.

In the second step, the blobs are analyzed in order to extract a noisy measurement for the position of each dancer. If a single dancer is associated with a blob, then that dancer's position measurement is taken as the centroid of all pixels in that blob. Otherwise, to resolve multi-dancer blobs or clusters of densely-spaced blobs, an expectation-maximization mixture-of-gaussian (EMMG) algorithm is used, which iteratively adjusts dancer positions for a given cluster and provides position measurements as output.

In the third step, the noisy position measurements are used with an unscented Kalman filter (UKF) for each dancer to provide a more accurate estimated position $(x, y)$ in the current frame and to predict the position in the next frame. The estimated position of each dancer is stored as the current point in the dancer's tracked trajectory. The predicted positions are used to inform the next tracking iteration. The $(x, y)$ position vector is expressed in a coordinate frame that is parallel to the floor. The transformation to these coordinates from image plane coordinates was determined by applying camera calibration techniques to an image of several objects placed at known locations in the scene. The average height of each dancer is assumed to be 1.65 meters.

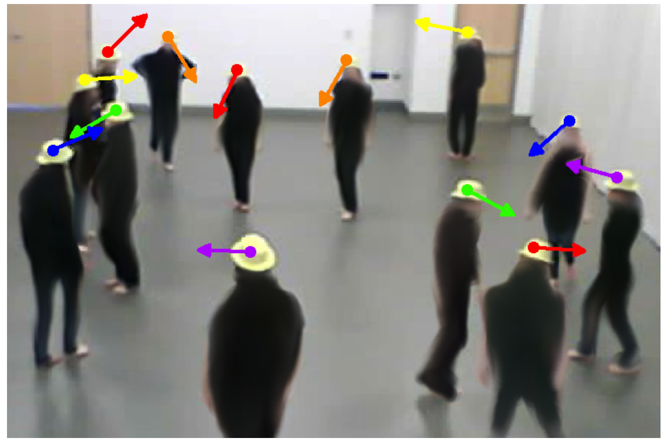

Fig. 1. One frame from the video of the experiment with superimposed tracked positions (colored dots) and normalized velocity vectors (colored arrows). Images of dancers are are deliberately blurred.

## IV. GRAPH THEORY AND VISUALIZATION

### A. Background on Graphs

Let $N$ be the number of dancers. For each dancer $i$ we define the set of neighbors, $\mathcal{N}_i$, to be the set of dancers whose positions are observed and used for cohesion by dancer $i$.

We associate to the system a *sensing graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$, where $\mathcal{V} = \{1, 2, \ldots, N\}$ is the set of nodes, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges and $A$ is the $N \times N$ adjacency matrix with $a_{i,j} = 1$ when edge $(i, j) \in \mathcal{E}$ and $a_{i,j} = 0$ otherwise. Every node in the graph corresponds to a dancer, while the graph contains edge $(i, j)$ when $j \in \mathcal{N}_i$. An edge $(i, j) \in \mathcal{E}$ is said to be *undirected* if $(j, i)$ is also in $\mathcal{E}$; otherwise it is *directed*. A graph is undirected if every edge is undirected, that is, if $A$ is symmetric; otherwise it is directed.

A graph can be represented visually by drawing a dot for each node and a line between the appropriate pair of nodes for each edge. An undirected edge is typically drawn as a simple line, while a directed edge $(i, j)$ will have an arrow head pointing from node $i$ to node $j$.

A *path* in $\mathcal{G}$ is a (finite) sequence of nodes containing no repetitions and such that each node is a neighbor of the previous one. The length of a path is given by the number of edges traversed by the path. The *distance*, $d_{i,j}$, between nodes $i$ and $j$ in a graph is the shortest length of any path from $i$ to $j$. If no such path exists, $d_{i,j}$ is infinite.

The graph $\mathcal{G}$ is *connected* if it contains a globally reachable node $k$; i.e. there is a path in $\mathcal{G}$ from $i$ to $k$ for every node $i$. $\mathcal{G}$ is said to be *strongly connected* if there is a path between every pair of nodes in the graph. A strongly connected component of $\mathcal{G}$ is a maximal subset of nodes such that there is a path in $\mathcal{G}$ between every pair of nodes in the subset. $\mathcal{G}$ is *weakly connected* if it is connected when every directed edge is replaced by an undirected edge. A weakly connected component is a maximal subset of nodes that forms a strongly connected component when every directed edge in $\mathcal{G}$ is replaced by an undirected edge.

The *status*, $s_k$, of a node $k$ is the average inverse distance between every other node and $k$. That is, $s_k =$

$\frac{1}{N-1}\sum_{j\neq k}\frac{1}{d_{j,k}}$. $s_k$ will be maximum (equal to 1) if there is an edge from every other node to node $k$, and minimum (equal to 0) if there are no edges leading to node $k$.

### B. Visualization of Graphs

FlockGrapher is our Matlab tool that computes, visualizes and evaluates different kinds of graphs derived from flock position data. Using a graphical user interface, the tool accepts tracked position and direction of motion data for individuals in a flock in two or three dimensions. It can visualize data from one specific instant in time or create a time series animation of data sets corresponding to successive time steps. The user can create graphs from the data by defining an individual's neighborhood in terms of either a prescribed number of nearest neighbors or a prescribed sensing radius. For data that includes the direction of motion of nodes, FlockGrapher can incorporate a limited viewing angle, assumed to be symmetric about the individual's direction of motion. In the case of a fixed number of nearest neighbors and a limited viewing angle, if there are fewer than the required number of neighbors visible to a node, the viewing angle will be rotated with respect to the direction of motion until enough neighbors are visible. Edge weights can be automatically manipulated, e.g., as a function of distance between nodes, or they can be prescribed by the user.

Once a sensing graph has been computed, FlockGrapher can evaluate a range of graph properties, including number of strongly and weakly connected components, algebraic connectivity, speed of convergence and node status. The tool also displays some properties on the graph visualization; for example, directed and undirected edges can be distinguished with different colors. For sets of data corresponding to successive time steps, the time-varying values of these properties will be displayed as the graph visualization changes. In the case of the human flocking experiment, this dynamic graph visualization can be run at the same time as the video of the dancers to compare computed and observed behavior. FlockGrapher can save all the computed data to allow for further analysis. A screenshot of FlockGrapher is shown in Figure 2; the graph and its properties corresponds to the frame from the video shown in Figure 1.

## V. SENSING MODEL AND GRAPH COMPUTATION

Since each dancer was given the same specific rules to follow, it is in principle possible to apply the same rules to our tracked data and reconstruct the sensing graph used by the dancers. However, certain aspects of both the rules and human behavior make this task challenging. Although the dancers were each told to stay arm's length from two other dancers, no instruction was given for how they were to choose these two neighbors. In addition, although humans have a field of view of up to $200°$ [17], there was no compulsion for the dancers to keep both of their neighbors visible at all times.

Given these limitations, we made two key assumptions in order to estimate the dancers' sensing graph. First we assumed that each dancer only chose neighbors from within a limited angular range centered about the direction they were travelling. Since no dancer was observed to be rapidly moving their head, the direction of motion is a reasonable proxy for direction of the head and therefore for center of viewing range. Second, we assumed that each dancer was applying the cohesion rule with the two *nearest* neighbors within this range. Since every dancer was trying to keep two neighbors at arm's length (and let no dancers closer than arm's length), a dancer's neighbors would naturally be among the closest of the other dancers.

With these assumptions we used FlockGrapher to estimate the sensing graph at each time (frame) by computing the two-nearest neighbor graph with a limited viewing angle. When fewer than two other dancers were visible using the direction of motion to center the viewing region, this region was allowed to rotate until two dancers became visible. However, we did not know *a priori* what viewing angle to choose to best represent the dancers' behavior.

For collective behavior, it is impossible to guarantee that a group will remain together if the communication graph is not connected [18]. When the graph is disconnected, there is nothing to prevent different subgroups from moving in different directions and splitting the group. However, other features of the environment (such as the limited space in the room) can drive the group back together. Since fissions and fusions of the group were observed, we selected the viewing angle for our sensing model as the one that produced a graph that was disconnected when the group of dancers split and remained connected when the group of dancers was cohesive.

Table I shows the results of estimating the sensing graph across the whole tracked period using different viewing angles. We can see that reducing the viewing angle from $360°$ to $270°$ significantly improves the amount of time the graph is connected, with the maximum connectedness occurring with a viewing angle of $120°$. However, our goal was not simply to maximize connectedness but rather to match the observed behavior of the group.

Early in the experiment, between about 1 and 3 seconds, a small group of four dancers split from the rest of the group. The dancers within this group appeared to be observing only one another. Eight of the remaining dancers also formed a group, only observing one another. The final dancer was able to observe both groups, but since no other dancer was observing this individual, the group remained split. Eventually, the dancers in the larger group turned and observed the smaller group, leading to a single "flock" again. This disconnection event was reflected in the estimated graphs for viewing angles of $150°$ and greater, but not for the smaller angles. However, with a viewing angle of $150°$ the graph became connected at a few points within this interval when direct observation of the video suggests that the group was still split. This was not the case with a viewing angle of $180°$; thus, $180°$ was chosen as providing the best match of the splitting behavior of the dancers. Figure 1 shows the group during this disconnection event and the graph in Figure 2 (corresponding to the frame of Figure 1) is computed using a viewing angle of $180°$.

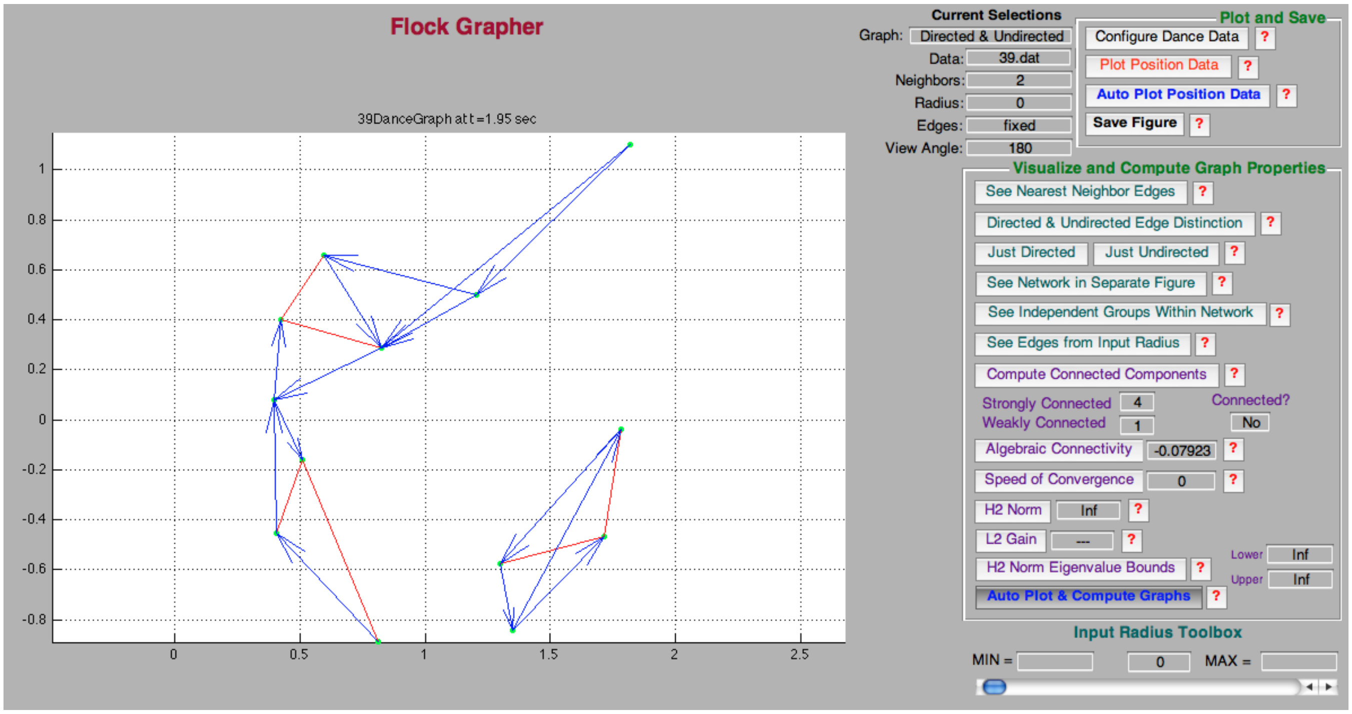


Fig. 2. Screenshot of FlockGrapher using dancer data corresponding to the instant shown in Figure 1. Nodes are shown as small green circles connected by edges. Directed edges are blue with arrow heads and undirected edges are red. Computed graph properties are displayed on the right.

TABLE I

EFFECTS OF VIEWING ANGLE ON GRAPH CONNECTEDNESS OVER THE WHOLE TRACKED PERIOD

| Total viewing angle | Percentage of time connected | Number of disconnection events |
|---|---|---|
| $360°$ | $59.58\%$ | 40 |
| $270°$ | $91.67\%$ | 43 |
| $210°$ | $97.5\%$ | 10 |
| $180°$ | $98.47\%$ | 3 |
| $150°$ | $98.68\%$ | 5 |
| $120°$ | $99.65\%$ | 3 |
| $90°$ | $99.58\%$ | 3 |

Although our estimate of the sensing graph captured a split in the group and remained connected during the rest of the tracked period, we acknowledge that it remains a crude estimate. For example, some nodes changed their neighbors rapidly in our estimated graph, which is likely an overestimation of the rate at which dancers switch neighbors. Instead, once a dancer chose a particular neighbor, it is likely that they maintained this neighbor for a period of time before switching. Thus, a more realistic sensing model might include a reluctance or "inertia" to change neighbors that have just been chosen.

## VI. ANALYSIS OF INDIVIDUAL INFLUENCE

We used the estimated time-varying sensing graph to begin investigating the influence of each individual within the group. Our method was to compute and compare node status. Without knowing precisely how each individual implemented the flocking rules, node status can provide an estimate of an individual's importance within the group. A dancer with a status of 0 has no influence since no one else in the group is observing that dancer. A dancer with a status of 1 has the maximum possible influence as every other individual is directly observing that dancer. However, due to the time-varying nature of the graph, an individual's importance depends not only on the current node status but also on its node status in the past. Therefore, as a first estimate of instantaneous importance we looked at each node's average status over the past 1 second. A plot of averaged node status for part of the tracked period is shown in Figure 3.

By examining our plot of averaged node status, we looked for "leadership events" where one particular node achieved the greatest importance within the group (with a high status value) for an extended period of time. In Figure 3 we can observe one such event when node 10 became a leader between approximately 28.75 and 31.45 seconds. Looking at the video, we observed that during this time the group was moving from the back left corner of the room toward the front right corner, with node 10 at the front of the group. This suggests that our node status measurements can capture emergent leadership.

Another leadership event was observed during a period when one dancer stopped moving and the remaining dancers started circling around this individual. However, the individual with the highest status during this event was not the stationary one, but one who was very close by and who kept moving in a circle. This seems particularly interesting since at other times (however not during our tracked period) one dancer would stop and the whole group would eventually stop too. The difference between these two kinds of events (circling versus stationary group motion) may be due to the differences between the status of the stationary and nearby dancers in the first case as compared to the second case.

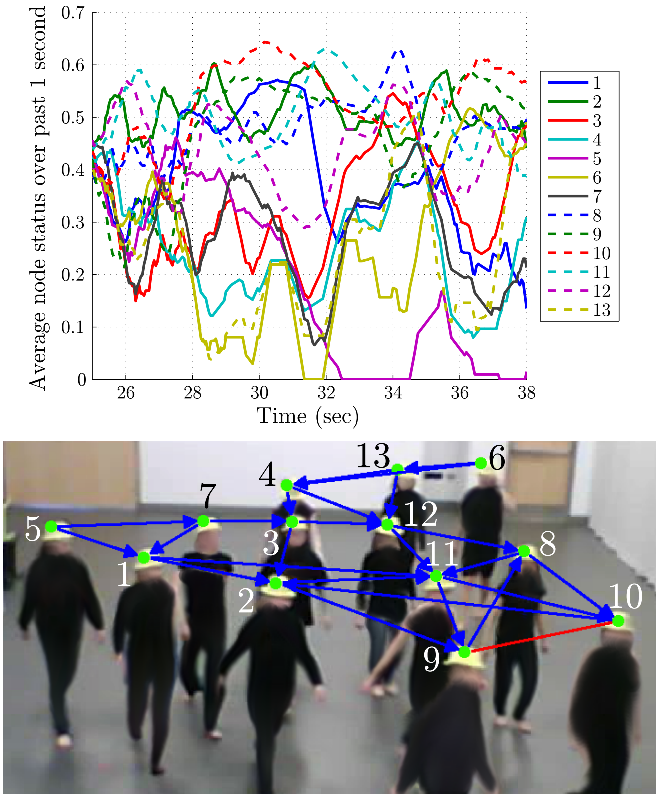


Fig. 3. Plot of 1-second running average of node status, along with a sample video frame and sensing graph near the end of the leadership event from $t = 28.75$ seconds to $t = 31.45$ seconds. The red edge is undirected while all blue edges are directed. We can observe that node 10, with the highest status, corresponds to the dancer in the front of the group.

By averaging each individual's status over the whole tracked period we investigated whether some individuals had a disproportionate influence on the group. Figure 4 shows the average of each node's status over the tracked period. We can see that nodes 12 and 10 had the highest average status, with values $1.9\sigma$ and $1.7\sigma$ higher than the group mean. This suggests that rather than leadership simply arising as a result of random mixing within the group, the behavior of some individuals makes them more likely to assume positions of high influence. We note that the dancers corresponding to nodes 10, 11 and 12 are three of the four dancers in the small disconnected group of Figures 1 and 2, suggesting further possible consequences of emergent leaders.

## VII. FINAL REMARKS

We described the artistic and scientific goals and methods for investigation in the Flock Logic project. To illustrate we tracked and analyzed the trajectories of thirteen dancers in a dance studio carrying out basic rules of flocking. We estimated the time-varying interconnection graph and computed node status; the results suggest emergence of leadership, where no such leadership was assigned. In ongoing work

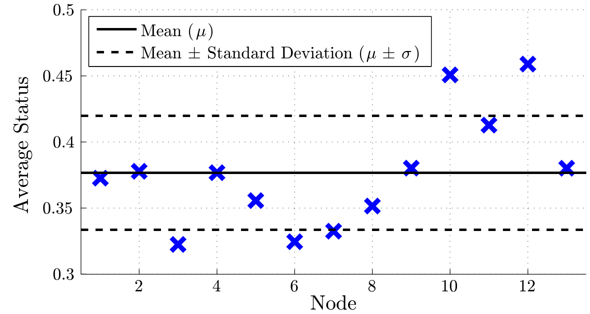


Fig. 4. Average node status over the whole tracked period.

we are comparing experimental results to simulated flocking to examine human bias. We are also computing correlations between leadership and shape, polarity and momentum of the group.

## REFERENCES


[1] N. E. Leonard and S. Marshall, "Flock Logic," 2010. [Online]. Available: www.princeton.edu/~flocklogic
[2] J. C. M. Breder, "Equations descriptive of fish schools and other animal aggregations," *Ecology*, vol. 35, no. 3, pp. 361–370, 1954.
[3] K. Clemente, "Playing with performance: The element of the game in experimental dance and theater," *The Journal of Popular Culture*, vol. XXIV, pp. 1–10, 1990.
[4] W. Forsythe, M. Palazzi, and N. Z. Shaw, "Synchronous objects," 2009. [Online]. Available: http://synchronousobjects.osu.edu/
[5] S. Sgorbati, "Emergent improvisation," 2005. [Online]. Available: http://emergentimprovisation.org/essay.html
[6] S. Gueron, S. A. Levin, and D. I. Rubenstein, "The dynamics of herds: From individuals to aggregations," *J. Theor. Biol.*, vol. 182, 1996.
[7] I. R. Fischhoff, S. R. Sundareson, J. Cordingley, and D. I. Rubenstein, "Habitat use and movements of plans zebra (equus burchelli) in response to predation danger from lions," *Behavioral Ecology*, vol. 18, no. 4, pp. 725–729, 2007.
[8] J. R. G. Dyer, A. Johansson, D. Helbing, I. D. Couzin, and J. Krause, "Leadership, consensus decision making and collective behaviour in humans," *Phil. Trans. R. Soc. B*, vol. 364, pp. 781–789, 2009.
[9] M. Moussaid, N. Perozo, S. Garnier, D. Helbing, and G. Theraulaz, "The walking behaviour of pedestrian social groups and its impact on crowd dynamics," *PLoS ONE*, vol. 5, no. 4, pp. 1–7, 2010.
[10] P. Funes, B. Orme, and E. Bonabeau, "Shaping collective behavior: an exploratory design approach," in *Artificial life IX: Proc. Ninth Int. Conf. on the Simulation and Synthesis of Artificial Life*, 2004.
[11] W. Chen, N. E. Leonard, and S. Marshall, "FlockMaker: Simulator for collective motion," 2010. [Online]. Available: dcsl.princeton.edu/dance
[12] D. T. Swain, I. D. Couzin, and N. E. Leonard, "Real-time feedback-controlled robotic fish for behavioral experiments with fish schools," *Proceedings of the IEEE*, in press 2011.
[13] D. T. Swain, "MADTraC framework source documentation," 2011. [Online]. Available: poincare.princeton.edu/MADTraC/docs/
[14] G. Bradski, "OpenCV (Open Computer Vision) library," March 2011. [Online]. Available: opencv.willowgarage.com/wiki/Welcome
[15] F. Chang, C.-J. Chen, and C.-J. Lu, "A linear-time component-labeling algorithm using contour tracing technique," *Computer Vision and Image Understanding*, vol. 93, no. 2, pp. 206–220, 2004.
[16] B. Mohar, "The Laplacian spectrum of graphs," *Graph theory, Combinatorics and Applications*, vol. 2, pp. 871–898, 1991.
[17] E. Werner and C. Rossi, *Manual of visual fields*. New York, NY: Churchill Livingstone, 1991.
[18] W. Ren, R. Beard, and E. Atkins, "A survey of consensus problems in multi-agent coordination," in *Proc. ACC*, 2005, pp. 1859–1864.