In *Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems* (AAMAS14), Paris, France, May 2014.

Orienting a Flock via Ad Hoc Teamwork

(Extended Abstract)

Katie Genter and Peter Stone
Department of Computer Science, University of Texas at Austin
Austin, TX 78712 USA
{katie,pstone}@cs.utexas.edu

ABSTRACT

Ad hoc teamwork refers to the challenge of designing agents that can influence the behavior of a team, without prior coordination with its teammates. This abstract summarizes a portion of our work on influencing a flock of agents to adopt a desired behavior within the context of ad hoc teamwork. We shortly summarize our work on examining how the ad hoc agents should behave in order to orient a flock towards a target heading as quickly as possible when given knowledge of, but no direct control over, the behavior of the flock. We overview three algorithms which the ad hoc agents can use to influence the flock, and summarize some of our initial results concerning the relative importance of coordinating the ad hoc agents versus planning farther ahead.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Multiagent systems

Keywords

Ad Hoc Teamwork; Coordination; Flocking

1. INTRODUCTION

In this abstract, we summarize some of our work on the problem of *leading* a team of flocking agents in an ad hoc teamwork setting. An ad hoc teamwork setting is one in which a teammate — which we call an *ad hoc agent* — must determine how to best achieve a team goal given a set of possibly suboptimal teammates. We assume that we control one or more ad hoc agents — perhaps in the form of robotic birds or ultralight aircraft — that are perceived by the rest of the flock as one of their own.

Flocking is an emergent behavior found in different species in nature including flocks of birds, schools of fish, and swarms of insects. In each of these cases, the animals follow a simple local behavior rule that results in a group behavior that appears well organized and stable. Specifically, we assume that each bird in the flock dynamically adjusts its heading based on that of its immediate neighbors.

We assume we are given a team of flocking agents following a known, well-defined rule characterizing their flocking behavior. In our research, we wish to examine how the ad hoc agents should behave. Specifically, our main research question is: *how should ad hoc agents behave so as to orient the rest of the flock towards a target heading as quickly as possible?*

Appears in: *Alessio Lomuscio, Paul Scerri, Ana Bazzan, and Michael Huhns (eds.), Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2014), May 5-9, 2014, Paris, France.*

2. PROBLEM DEFINITION

We use a simplified version of Reynolds' Boid algorithm for flocking [4] in which we assume that each agent calculates its orientation for the next time step to be the average heading of its *neighbors*. Specifically, an agent's *neighbors* are the agents located within some set radius of the agent. In order to calculate its orientation for the next time step, each agent computes the vector sum of the velocity vectors of each of its neighbors and adopts a scaled version of the resulting vector as its new orientation. At each time step, each agent moves one step in the direction of its current vector and then calculates its new heading based on those of its neighbors, keeping a constant speed.

Over time, agents behaving as described above will gather into one or more groups, and these groups will each travel in some direction. However, in our work we add a small number of *ad hoc agents* to the flock. These ad hoc agents attempt to influence the flock to travel in a pre-defined direction — we refer to this direction as θ^*. We conclude that the flock has converged to θ^* when every agent that is not an ad hoc agent is facing within 0.1 radians of θ^*.

3. 1-STEP LOOKAHEAD BEHAVIOR

In this section we provide a short overview of our 1-step lookahead algorithm for determining the individual behavior of each ad hoc agent. This behavior considers all of the influences on neighbors of the ad hoc agent, such that the ad hoc agent can determine the best orientation to adopt based on this information. The 1-step lookahead behavior is a greedy, myopic approach for determining the best individual behavior for each ad hoc agent, where 'best' is the behavior that will exert the most influence on the next time step.

The 1-step lookahead algorithm is called on each ad hoc agent at each time step. The output from the algorithm is the orientation that, if adopted by this ad hoc agent, is guaranteed to influence its neighbors to face closer to θ^* than any of the other *numAngles* discrete ad hoc orientations considered. Conceptually, our 1-step lookahead behavior is concerned with how the neighbors of each neighbor of the ad hoc agent are influenced if the ad hoc agent adopts a particular orientation at this time step.

4. 2-STEP LOOKAHEAD BEHAVIOR

Whereas the 1-step lookahead behavior optimizes each ad hoc agent's orientation to best influence its neighbors on the *next* step, it fails to consider more long-term effects. Hence, in this section we present a short overview of a 2-step lookahead behavior that considers influences on the neighbors of the neighbors of the ad hoc agent, such that the ad hoc agent can make a more informed decision when determining the best orientation to adopt.

Our 2-step lookahead algorithm is concerned with (1) how the neighbors of each neighbor of the ad hoc agent are influenced if the ad hoc agent adopts a particular orientation at this time step and (2) how the neighbors of the neighbors of each neighbor of the ad hoc agent are influenced if the ad hoc agent adopts a particular orientation at this time step, since they will influence the neighbors of each neighbor of the ad hoc agent on the next time step.

5. COORDINATED BEHAVIOR

The ad hoc agent behaviors above were for individual ad hoc agents, where each ad hoc agent calculated its behavior independent of any other ad hoc agents. We designed this behavior to determine whether ad hoc agents can exert more influence on the flock by working in coordinated pairs.

We select the ad hoc agents to pair by first finding all pairs of ad hoc agents with one or more neighbors in common. Then we do a brute-force search and find every possible disjoint combination of these pairs. For each such combination, we calculate the sum of the number of shared neighbors across all the pairs and select the combination with the greatest sum of shared neighbors.

The behavior of each ad hoc agent depends on whether it is part of a pair or not. If it is part of a pair, it follows our coordinated algorithm. If it is not part of a pair, it performs a 1-step lookahead search for the best individual behavior. Our coordinated algorithm considers each of the *numAngles* ad hoc agent orientations for the ad hoc agent and for the ad hoc agent's partner and essentially performs two 1-step lookahead searches.

6. EXPERIMENTS

In this section we summarize one of our experiments testing the three ad hoc agent behaviors discussed above against two baseline methods.

6.1 Baseline Ad Hoc Agent Behaviors

The first baseline method we consider is *Face Desired Orientation Behavior*, which we modeled after work by Jadbabaie, Lin, and Morse [2]. When following this behavior, the ad hoc agents always orient towards θ^*. Under this behavior the ad hoc agents do not consider their neighbors or anything about their environment when determining how to behave.

The second baseline method we consider is *Offset Momentum Behavior*, which was inspired by our previous work [1] in which we showed how to optimally orient a stationary agent to a desired orientation using a set of stationary ad hoc agents. Under this behavior, each ad hoc agent calculates the vector sum V of the velocity vectors of its neighbors and then adopts an orientation along the vector V' such that the vector sum of V and V' points towards θ^*.

6.2 Experimental Setup

We utilize the MASON simulator [3] for the experiment summarized in this abstract. We use a simplified version of Reynolds' Boid algorithm for flocking, so weight is only given to the 'Consistency' vector in the MASON simulator. We use the default simulator setting of 150 units for the height and width of our domain and we use the default setting in which each agent moves 0.7 units during each time step.

The number of agents in our simulation (*numAgents*) is 200, meaning that there are 200 agents in our flock. 10% of the flock are ad hoc agents. The neighborhood for each agent is 20 units in diameter. *numAgents* and the neighborhood size were both default values for MASON. We only consider *numAngles* discrete angle choices for each ad hoc agent. In the experimental results summarized below, *numAngles* is 50, meaning that the unit circle is equally divided into 50 segments beginning at 0 radians and each of these orientations is considered as a possible orientation for each ad hoc agent.

We ran 50 trials for each experimental setting and used the same 50 random seeds for each set of experiments for the purpose of variance reduction.

6.3 Experimental Results

In this abstract we summarize a set of results obtained under one experimental setup. Table 1 shows the number of time steps needed for the flock to converge to θ^* for the two baseline algorithms, the 1-step lookahead algorithm, the 2-step lookahead algorithm, and the coordinated algorithm using the experimental setup described in Section 6.2.

Algorithm	Time Steps	95% CI ($\pm$)
Face Desired Orientation Behavior	34.82	3.85
Offset Momentum Behavior	36.70	4.63
1-Step Lookahead Behavior	26.02	3.10
2-Step Lookahead Behavior	25.94	3.16
Coordinated Behavior	25.76	3.15

Table 1: The number of time steps required for the flock to converge to θ^* using the experimental setup described in Section 6.2. CI stands for confidence interval.

The results shown in Table 1 clearly show that the 1-Step Lookahead Behavior, the 2-Step Lookahead Behavior, and the Coordinated Behavior all perform significantly better than the two baseline methods under this experimental setting. However, these results did not show the 2-Step Lookahead Behavior and the Coordinated Behavior performing significantly better than the 1-Step Lookahead Behavior for this experimental setting.

7. REFERENCES

[1] K. Genter, N. Agmon, and P. Stone. Ad hoc teamwork for leading a flock. In *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2013)*, May 2013.

[2] A. Jadbabaie, J. Lin, and A. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48(6):988 – 1001, June 2003.

[3] S. Luke, C. Cioffi-Revilla, L. Panait, K. Sullivan, and G. Balan. Mason: A multi-agent simulation environment. *Simulation: Transactions of the Society for Modeling and Simulation International*, 81(7):517–527, 2005.

[4] C. W. Reynolds. Flocks, herds and schools: A distributed behavioral model. *SIGGRAPH*, 21:25–34, August 1987.