COOPERATIVE PREY HERDING BASED ON DIFFUSION ADAPTATION

Sheng-Yuan Tu and Ali H. Sayed

Department of Electrical Engineering
University of California, Los Angeles, CA 90095

ABSTRACT

Mobile adaptive networks consist of a collection of nodes with learning and motion abilities that interact with each other locally in order to solve distributed processing and distributed inference problems in real-time. In this paper, we develop adaptation algorithms that exhibit self-organization properties and apply them to the model of cooperative hunting among predators. The results help provide an explanation for the agile adjustment of network patterns in the interaction between fish schools and predators.

Index Terms— Cooperative hunting, self-organization, diffusion adaptation, mobile adaptive networks.

1. INTRODUCTION

Self-organization in biological networks emerges from the localized interactions among the members of the network [1, 2]. One interesting organized behavior in animal groups is their collective motion, where animals move together in amazing synchrony such as fish schools swimming together [3], bees swarming towards a hive, or birds flying in V-formations [4].

In fish schools, the individual members tend to move coherently while avoiding collisions. Such schooling behavior helps fish discourage attacks from their predators [5]. In [6], we used the concept of adaptive networks, along with diffusion adaptation mechanisms [7–9], to model the schooling behavior of fish and how they avoid mobile predators.

Cooperative behavior can be observed among predators as well. For example, dolphins encircle their prey [10] and killer whales cooperatively herd herring into a tight ball close to the surface [11]. Cooperative hunting plays a role in increasing foraging efficiency. In this paper, we use diffusion adaptation to explain how predators cooperate with each other to surround a fish school and trap the school while attacking.

2. DISTRIBUTED ESTIMATION

2.1. Measurement Model

Let w° denote the location vector of a target that the fish school wishes to track (e.g., the location of a food source). As Fig. 1 shows, the distance $d_k^\circ(i)$ between the target and node k at location $x_{k,i}$ at time i is given by the inner product

$$d_k^\circ(i) = u_{k,i}^\circ(w^\circ - x_{k,i}) \tag{1}$$

This work was supported in part by NSF grants CCF-1011918, CCF-0942936, and ECS-0725441. Authors' emails:{shinetu, sayed}@ee.ucla.edu.

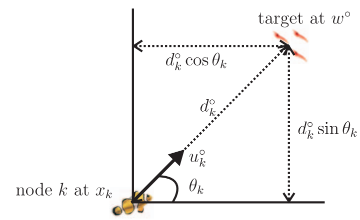

Fig. 1. Distance and direction of the target w° from node k at location x_k. The unit direction vector u_k° points towards w°.

where $u_{k,i}^\circ$ denotes the unit (row) direction vector pointing to w° from $x_{k,i}$; this vector is defined in terms of the azimuth angle, $\theta_k(i)$, i.e.,

$$u_{k,i}^\circ = \begin{bmatrix}\cos\theta_k(i) & \sin\theta_k(i)\end{bmatrix} \tag{2}$$

The superscript $\circ$ in (1)-(2) is used to indicate true values. However, nodes observe noisy measurements of the direction $u_{k,i}^\circ$ and the distance $d_k^\circ(i)$ to the target, say,

$$u_{k,i} = u_{k,i}^\circ + n_{k,i}^u, \quad d_k(i) = d_k^\circ(i) + n_k^d(i) \tag{3}$$

where $n_{k,i}^u$ and $n_k^d(i)$ denote additive noise terms of sizes M and 1, respectively. Rearranging the above equations, we obtain a linear regression model relating $d_k(i)$ and $u_{k,i}$ to w°, namely,

$$d_k(i) + u_{k,i}x_{k,i} = u_{k,i}w^\circ + n_k(i) \tag{4}$$

where the scalar noise term $n_k(i)$ is given by

$$n_k(i) \triangleq -n_{k,i}^u(w^\circ - x_{k,i}) + n_k^d(i)$$

2.2. Diffusion Adaptation

Consider a set of N nodes distributed over some spatial region. Two nodes are said to be neighbors if they can share information (i.e., exchange some data). The set of neighbors of node k, including node k itself, is called the neighborhood of k and is denoted by $\mathcal{N}_k$. The objective of the network is to estimate w° in a fully distributed manner and in real-time, where each node is allowed to interact only with its neighbors. One such scheme is the so-called Adapt-then-Combine (ATC) diffusion algorithm [9, 12]. The algorithm is a stochastic gradient solution that optimizes a mean-square error cost function in a fully distributed manner (derivations and mean-square-error analysis can be found in [9]). The algorithm is described as follows:

$$\boxed{\begin{aligned}\psi_{k,i} &= w_{k,i-1} + \mu_k u_{k,i}^T[d_k(i) - u_{k,i}(w_{k,i-1} - x_{k,i})] \\ w_{k,i} &= \sum_{j\in\mathcal{N}_k} a_{j,k}\psi_{j,i}\end{aligned}} \tag{5}$$

where μ_k is a positive step size used by node k and $\{a_{j,k}\}$ is a set of non-negative real weights assigned to node k and satisfying:

$$\sum_{j=1}^{N} a_{j,k} = 1, \quad a_{j,k} = 0 \text{ if } j \notin \mathcal{N}_k \tag{6}$$

The resulting estimate of node k at time i is denoted by $w_{k,i}$. In implementation (5), the nodes in the neighborhood of node k share their intermediate estimates $\{\psi_{j,i}\}$.

In the application we are studying in this paper, the nodes of the network wish to track two separate targets: the location of the food source and the location of the predator. The modeling equations described above apply to either target. To distinguish between them, we shall use superscripts f and p for food and predator, respectively. Thus, instead of w°, we shall write w^f for the actual location of the food source and w^p for the actual location of the predator. In addition, variables without superscripts will denote quantities that are related to the nodes of the adaptive network.

3. MOTION CONTROL MECHANISM FOR ADAPTIVE NETWORKS (FISH SCHOOLS)

Before we proceed, we introduce two operators on 2×1 vectors. Let $v = \begin{bmatrix} v_1 & v_2 \end{bmatrix}^T$ be a 2×1 vector. Then we define

$$\mathfrak{u}(v) \triangleq v/\|v\|; \quad v^\perp \triangleq \begin{bmatrix} -v_2 & v_1 \end{bmatrix}^T$$

That is, $\mathfrak{u}(v)$ normalizes the vector and $\perp$ finds a vector perpendicular to v.

In a mobile network, every node k updates its location vector over time according to the rule:

$$x_{k,i+1} = x_{k,i} + \triangle t \cdot v_{k,i+1} \tag{7}$$

where $\triangle t$ represents the time step and $v_{k,i+1}$ is the velocity vector of the node. As was shown in [6], there are three factors influencing the velocity vector of node k. First, the action of each node depends on the location of the predator. Referring to Fig. 2, each node focuses on foraging in region I and on escaping from predators in the other regions. The velocity vector is set as follows:

$$v^a_{k,i+1} = \begin{cases} \mathfrak{u}(w^f_{k,i} - x_{k,i}) & \text{(region I)} \\ c_1 \cdot \mathfrak{u}[(x_{k,i} - w^p_{k,i})^\perp] & \text{(region II)} \\ -\mathfrak{u}(v^p_{k,i}) & \text{(region III)} \\ \left(2r_1/\|x_{k,i} - w^p_{k,i}\| - 1\right) \mathfrak{u}(x_{k,i} - w^p_{k,i}) & \text{(region IV)} \end{cases} \tag{8}$$

where c_1 is equal to 1 if the inner product

$$(x_{k,i} - w^p_{k,i})^T (v^p_{k,i})^\perp$$

is greater than zero; otherwise, c_1 is equal to -1. Here, we use $w^f_{k,i}$ and $w^p_{k,i}$ to denote the local estimates at node k at time i. For multiple predators, each node in the fish school tracks the location of the nearest predator. The estimation is implemented by algorithm (5). Moreover, in (8), $v^p_{k,i}$ is the local estimate of the predator's velocity, which can be estimated as

$$\boxed{v^p_{k,i} = \frac{1}{\triangle t}(w^p_{k,i} - w^p_{k,i-1})} \tag{9}$$

Second, a network may become fragmented after an attack by a predator. To reunite, nodes on the outer boundaries have to estimate

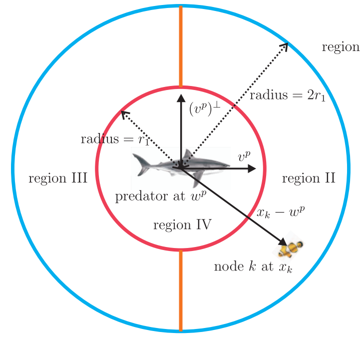

Fig. 2. Two concentric circles with the origin at the predator and radii r_1 and $2r_1$. The four regions represent the areas outside the circle of radius $2r_1$, inside the circle of radius r_1, and in front and behind the predator within the disc $r_1 < r < 2r_1$.

the location of the other groups and move towards them. Let $x_{j,i}$ denote the location of the nearest node in the other fragment. The velocity vector is then set as:

$$v^b_{k,i+1} = \begin{cases} 0, & \text{if no other group is found} \\ \mathfrak{u}(x_{j,i} - x_{k,i}), & \text{otherwise} \end{cases} \tag{10}$$

Finally, the nodes want to move in synchrony to confuse predators and would like to avoid collisions by maintaining a safe distance r from their neighbors. This can be achieved if the node updates its velocity vector as follows [13]:

$$v^c_{k,i+1} = v^g_{k,i} + \gamma \delta_{k,i} \tag{11}$$

where γ is a nonnegative scalar and $\delta_{k,i}$ deals with collision avoidance, which is given by

$$\delta_{k,i} = \frac{1}{|\mathcal{N}_k| - 1} \sum_{j \in \mathcal{N}_k \backslash \{k\}} \left(\|x_{j,i} - x_{k,i}\| - r\right) \mathfrak{u}(x_{j,i} - x_{k,i}) \tag{12}$$

Expression (11) also incorporates the term $v^g_{k,i}$, which refers to a local estimate for the velocity of the center gravity of the network and is estimated in a distributed manner as follows:

$$\boxed{\begin{aligned} \varphi_{k,i} &= (1 - \mu^v_k) v^g_{k,i-1} + \mu^v_k v_{k,i} \\ v^g_{k,i} &= \sum_{j \in \mathcal{N}_k} a^v_{j,k} \varphi_{j,i} \end{aligned}} \tag{13}$$

According to the criteria (8)-(13), we propose the following mechanism by which $v_{k,i+1}$ can be set by node k; this mechanism is an extension of the one proposed in [13] where we are now adding a new component $v^a_{k,i+1}$ to help avoid predators as defined by (8):

$$\boxed{v_{k,i+1} = \lambda(\alpha v^a_{k,i+1} + \beta v^b_{k,i+1}) + (1 - \lambda) v^g_{k,i} + \gamma \delta_{k,i}} \tag{14}$$

where $\{\lambda, \alpha, \beta, \gamma\}$ are non-negative weighting factors. We bound the maximum speed of nodes by v_{max} so that the magnitude of $v_{k,i+1}$ will be scaled to v_{max} if it is larger than v_{max}.

4. MOTION CONTROL MECHANISM FOR PREDATORS

Now, consider M predators that would like to hunt the fish nodes in the network cooperatively. The location of predator l at time i is

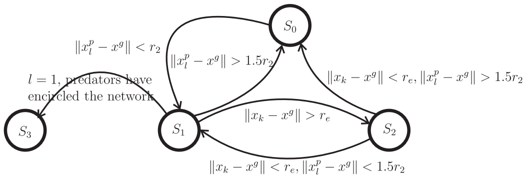

Fig. 3. State transition diagram of the motion model of predators.

denoted by $x_{l,i}^p$. The predators coordinate their behavior in order to improve their hunting efficiency. The motion of the predators is influenced by two quantities related to the network: the location of the center gravity of the network and the location of a node of interest. Let $x_{l,i}^g$ and $w_{l,i}$ denote the estimated locations of the network center and the node of interest by predator l at time i. Predators cooperatively estimate $x_{l,i}^g$ by using (5). However, since predators spread around the network, they have different nodes of interest. Therefore, each predator has to track $w_{l,i}$ independently by

$$w_{l,i} = w_{l,i-1} + \nu u_{l,i}^T [d_l(i) - u_{l,i}(w_{l,i-1} - x_{l,i}^p)] \tag{15}$$

where ν is a step size, and $u_{l,i}$ and $d_l(i)$ are the measured direction and distance of the node, respectively, at predator l and time i.

4.1. State Machine Model

We model the behavior of predators as a finite-state machine with four possible states, S_0 to S_3. The state transition diagram is depicted in Fig. 3. Predator l initially enters state S_0 and moves towards the network (i.e., fish school) until it is close to the network, say, $\|x_{l,i}^p - x_{l,i}^g\| < r_2$. Then the predator moves to state S_1 and tries to encircle the network by moving around it. The predator monitors the node that is within a distance r_s and is the farthest from the network center. If the node is far away from the network center (i.e., an outlier), say, $\|w_{l,i} - x_{l,i}^g\| > r_e$, the predator enters state S_2 and drives the node back until it is within the distance r_e from the network center. After that, the predator may go back to state S_1 or S_0 depending on the distance to the network center. If the distance is greater than $1.5r_2$, the predator moves to state S_0; otherwise it moves to state S_1. Similarly, the predator in state S_1 may transit to state S_0 if it is far away from the center, i.e., $\|x_{l,i}^p - x_{l,i}^g\| > 1.5r_2$. Finally, after predators have encircled the network, predators take turns to attack the network. We assume that only one predator, say predator 1, will launch an attack to focus on the agile adjustment of network patterns in the network and predators.

4.2. State S_0: Chase

Chasing happens when the distance to the network center is large. To get closer to the network, the predator sets the velocity vector towards the network center, i.e.,

$$v_{l,i+1}^d = \mathfrak{u}(x_{l,i}^g - x_{l,i}^p) \tag{16}$$

4.3. State S_1: Encircle

In S_1, predators would like to encircle the network within a disc with the origin at the network center and radius r_e by moving around the network center. In addition, predators would like to distribute evenly around the network in order to make it difficult for the nodes in the network to escape. To avoid staying together, predator l first checks if there are other predators within distance r_s. If yes, say predator j, predator l then determines the direction of predator j by the inner product $(x_{j,i}^p - x_{l,i}^g)^T (x_{l,i}^p - x_{l,i}^g)^\perp$. We say predator j lies in the right semicircle of predator l if the inner product is greater than zero (see Fig. 4). Predator l sets the velocity vector towards the empty semicircle; otherwise, predator l randomly chooses a direction, i.e.,

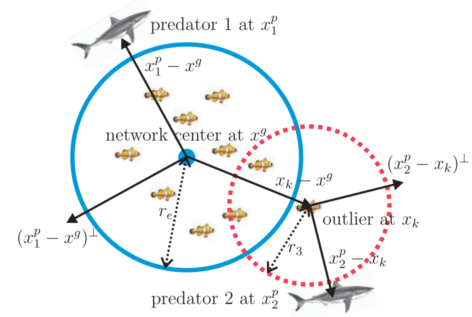

Fig. 4. Location relations between the network and predators.

$$v_{l,i+1}^d = c_2 \cdot \mathfrak{u}[(x_{l,i}^p - x_{l,i}^g)^\perp] \tag{17}$$

where c_2 determines the direction and is equal to 1 or -1 (i.e., counterclockwise or clockwise). If the right semicircle is empty, $c_2 = 1$. Similarly, $c_2 = -1$ if the left semicircle is empty. Otherwise, c_2 is equally likely to be -1 or 1.

4.4. State S_2: Trap

Outliers occur when they try to escape from an attack. To push the outlier back, the predator moves to the front of the outlier and blocks it. However, if the predator directly approaches the outlier, it may move further away to escape from the predator. To avoid this situation, the predator keeps a certain distance to the outlier and moves around the outlier until it blocks the way out. To do so, the predator sets the velocity vector as follows:

$$v_{l,i+1}^d = \begin{cases} \mathfrak{u}(w_{l,i} - x_{l,i}^p) & \text{if } \|w_{l,i} - x_{l,i}^p\| > 1.5r_3 \\ -\mathfrak{u}(w_{l,i} - x_{l,i}^p) & \text{if } \|w_{l,i} - x_{l,i}^p\| < r_3 \\ c_3 \cdot \mathfrak{u}[(w_{l,i} - x_{l,i}^p)^\perp] & \text{otherwise} \end{cases} \tag{18}$$

where c_3 is equal to 1 if the inner product $(w_{l,i} - x_{l,i}^g)^T (x_{l,i}^p - w_{l,i})^\perp$ is greater than zero; otherwise, c_3 is equal to -1.

4.5. State S_3: Attack

When attacking, the predator tracks the location and velocity of the nearest node and moves towards the predicted location of that node. That is, the velocity vector of the predator is updated as:

$$v_{l,i+1}^d = \mathfrak{u}(w_{l,i} + \triangle t \cdot v_{l,i} - x_{k,i}^p) \tag{19}$$

where $v_{l,i}$ is the estimated velocity of the node by predator l at time i and is estimated in the same way as (9).

Predators also avoid collisions and will move apart when they are too close. The final adjustment of the velocity and location vectors by predator l is as follows:

$$\boxed{\begin{aligned} v_{l,i+1}^p &= \lambda^p v_{l,i+1}^d + \gamma^p \delta_{l,i}^p \\ x_{l,i+1}^p &= x_{l,i}^p + \triangle t \cdot v_{l,i+1}^p \end{aligned}} \tag{20}$$

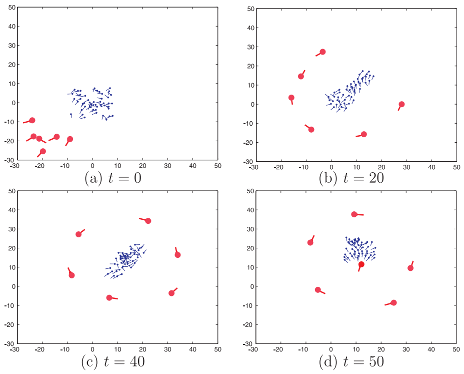

Fig. 5. A simulation showing how predators coordinate their behavior to encircle a fish school. The behavior of the fish school and the predators are modeled using diffusion adaptation over networks.

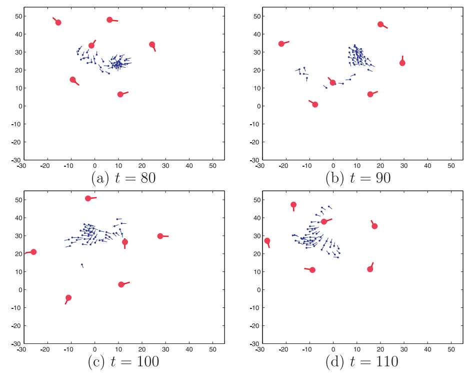

Fig. 6. Attacking behavior of predators over time.

where $\{\lambda^p, \gamma^p\}$ are non-negative weighting scalars and

$$\delta^p_{l,i} = \frac{1}{|\mathcal{M}_l| - 1} \sum_{j \in \mathcal{M}_l \setminus \{k\}} \left(r_p - \|x^p_{l,i} - x^p_{j,i}\| \right) \mathfrak{u}(x^p_{l,i} - x^p_{j,i}) \tag{21}$$

In (21), $\mathcal{M}_l$ is a set of predators within distance r_p, i.e., $\mathcal{M}_l = \{j : \|x^p_{l,i} - x^p_{j,i}\| < r_p\}$. Moreover, the speed of predators is bounded by v^p_{max}.

5. SIMULATION RESULTS

In this section, we simulate the motion of 50 nodes and 6 predators. The simulation parameters are set as follows. The unit length is the body length of a node (e.g., body length of a fish). All step sizes are set to 0.5. The combination coefficients are set to $a_{l,k} = a^v_{l,k} = 1/|\mathcal{N}_{k,i}|$ if $l \in \mathcal{N}_{k,i}$. For velocity control, the coefficients are $(\lambda, \alpha, \beta, \gamma) = (0.5, 1, 2, 1)$ and $(\lambda^p, \gamma^p) = (2.4, 1)$. The maximum speeds are $v_{max} = 2$ and $v^p_{max} = 2.4$, and the time duration is $\triangle t = 0.5$ sec. In addition, the distance parameters are set to $r = 3$, $r_p = 5$, $r_s = 30$, $r_e = 15$ and $(r_1, r_2, r_3) = (10, 20, 15)$.

We illustrate the maneuver of a mobile network in $\mathbb{R}^2$ over time in Fig. 5 and 6. The blue symbols "•" and "−" indicate the locations and moving directions of the nodes, respectively. The red symbols with bigger sizes represent predators. Figure 5 shows that predators encircle the network in the beginning and then one predator launches an attack. In Fig. 6, we observe the predators trapping the network while one predator is attacking. The simulation results emulate the behavior of fish schools in nature.

6. CONCLUSIONS

In this paper, we proposed a diffusion adaptation model to emulate the interactive behavior between fish schools and predators. The algorithm is implemented in a fully distributed and adaptive manner. The algorithm helps explain how fish schools avoid attacks from predators and how predators cooperatively hunt prey.

7. REFERENCES

[1] S. Camazine, J. L. Deneubourg, N. R. Franks, J. Sneyd, G. Theraulaz, and E. Bonabeau, *Self-Organization in Biological Systems*, Princeton University Press, 2003.

[2] I. D. Couzin, "Collective cognition in animal groups," *Trends in Cognitive Sciences*, vol. 13, pp. 36–43, Jan. 2009.

[3] S. Y. Tu and A. H. Sayed, "Foraging behavior of fish schools via diffusion adaptation," *Proc. International Workshop on Cognitive Information Processing*, pp. 63–68, Elba Island, Italy, Jun. 2010.

[4] F. Cattivelli and A. H. Sayed, "Self-organization in bird flight formations using diffusion adaptation," *Proc. International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, pp. 44–52, Aruba, Dutch Antilles, Dec. 2009.

[5] G. Turner and T. Pitcher, "Attack abatement: A model for group protection by combined avoidance and dilution," *American Naturalist*, vol. 128, pp. 228–240, 1986.

[6] S. Y. Tu and A. H. Sayed, "Tracking behavior of mobile adaptive networks," *Proc. Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, Nov. 2010.

[7] C. G. Lopes and A. H. Sayed, "Incremental adaptive strategies over distributed networks," *IEEE Trans. on Signal Processing*, vol. 55, pp. 4064–4077, Aug. 2007.

[8] C. G. Lopes and A. H. Sayed, "Diffusion least-mean squares over adaptive networks: Formulation and performance analysis," *IEEE Trans. on Signal Processing*, vol. 56, pp. 3122–3136, Jul. 2008.

[9] F. S. Cattivelli and A. H. Sayed, "Diffusion LMS strategies for distributed estimation," *IEEE Trans. on Signal Processing*, vol. 58, pp. 1035–1048, Mar. 2010.

[10] K. J. Benoit-Bird and W. W. L. Au, "Cooperative prey herding by the pelagic dolphin, stenella longirostris," *Journal of Acoustical Society of America*, vol. 125, pp. 125–137, Jan. 2009.

[11] T. Simila and F. Ugarte, "Surface and underwater observations of cooperatively feeding killer whales in northern Norway," *Canadian Journal of Zoology*, vol. 71, pp. 1494–1499, 1993.

[12] N. Takahashi, I. Yamada, and A. H. Sayed, "Diffusion least-mean squares with adaptive combiners: Formulation and performance analysis," *IEEE Trans. Signal Processing*, vol. 8, pp. 4795–4810, Sep. 2010.

[13] S. Y. Tu and A. H. Sayed, "Mobile adaptive networks with self-organization abilities," *Proc. International Symposium on Wireless Communication Systems*, pp. 379–383, York, United Kingdom, Sep. 2010.