Asian Journal of Control, Vol. 10, No. 2, pp. 212–232, March 2008
Published online in Wiley InterScience (www.interscience.wiley.com) DOI: 10.1002/asjc.020

COMPARISON OF THREE ORIENTATION AGREEMENT STRATEGIES IN SELF-PROPELLED PARTICLE SYSTEMS WITH TURN ANGLE RESTRICTIONS IN SYNCHRONOUS AND ASYNCHRONOUS SETTINGS

Andaç T. Samiloglu, Veysel Gazi, and A. Bugra Koku

ABSTRACT

In this study, we compare three different orientation agreement strategies of multi-agent/particle systems under different conditions. We investigate the behavior of multi-agent systems utilizing these strategies with different combinations of the following properties: (i) the multi-agent systems may be synchronous or asynchronous, (ii) they may travel in bounded or unbounded regions and (iii) the mobile agents may have turning speed restrictions. The agents/particles are assumed to move with constant speed and update their orientation of motion based on three different strategies. Based on these strategies, simulations are performed and the effects on the clustering performance are investigated.

Key Words: Consensus, distributed agreement, orientation agreement, asyncronism, turn angle restrictions, multi-agent systems.

I. INTRODUCTION

The collective motion of organisms like schools of fish, herds of quadrupeds, flocks of flying birds, and groups of migrating bacteria, molds, ants, or pedestrians is an interesting area studied by many biologists, physicists, and even engineers in recent years. The coordinated behavior of such animal groups results in complex and meaningful emergent or self-organizing behavior with only local interactions of relatively simple or "dumb" individuals (or agents as we call them here). Life sciences like theoretical biology and animal ethology can benefit from the ideas or principles derived from the operation of natural multi-agent systems. The developed ideas and principles may also be utilized in many engineering fields including swarm robotics [1, 2], optimization [3–7], self-organizing distributed sensor networks [8], decentralized/distributed coordination and control of groups of unmanned air, space, land and underwater vehicles, or even problems of social sciences including organization theory, economics, and cognitive psychology. Hence, for several decades many scientists from different fields have been trying to understand, model, and mimic/reproduce the behavior seen in natural swarms.

Among the first relevant works by biologists are the studies by Breder [9], Warburton and

Manuscript received April 30, 2007; revised November 5, 2007; accepted January 15, 2008.

Andaç T. Samiloglu is with the Baskent University Department of Mechanical Engineering Bağlıca Kampüsü Eskişehir Yolu 20.Km Bağlıca 06810 ANKARA. He is a Ph.D. student in METU, Mechanical Engineering Department and also graduate research associate at TOBB University of Economics and Technology (e-mail: andacsam@baskent.edu.tr).

Veysel Gazi is with the Dept. of Electrical and Electronics Engineering, TOBB University of Economics and Technology (ETU), Söğütözü Cad. No: 43, 06560 Ankara, TURKEY (e-mail: vgazi@etu.edu.tr).

A. Bugra Koku is with Middle East Technical University (METU), Mechanical Engineering Department, İnönü Bulvarı, Çankaya, Ankara, TURKEY (e-mail: kbugra@metu.edu.tr).

This work was supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK) under grant 104E170.

Lazarus [10], Okubo and Grunbaum [11–13], and Parrish [14].

The first study on simulating of flocking behavior of birds was performed by Reynolds in his well-known study [15], where he showed that, if followed by the simulated agents, three simple rules can result in realistic behavior similar to the one observed in bird flocks. The model of self-propelled particles considered by Vicsek [16] is similar in nature to the model of Reynolds, except that the particles in the Vicsek's model have constant speed. In that work, they considered a self-propelled particle system with dynamics based on the simple rule "at each time step a given particle driven with a constant absolute velocity assumes the average direction of motion of the particles in its neighborhood of radius r with some random perturbation added" [17], and investigated clustering, transport, and phase transition in non-equilibrium systems. They showed that their model results in a rich/realistic dynamics despite the simplicity of the model. In [18] and [19] Czirók *et al.* study biologically inspired, inherently non-equilibrium models consisting of self-propelled particles. Similar to [17], the particles move on a plane with constant speed and interact with their neighbors by choosing, at each time step, a heading equal to the average direction of their neighbors. In [18], they showed that the far-from-equilibrium system of self-propelled particles can be described using the framework of classical critical phenomena and the analysis shows new features when compared with the analogous equilibrium systems. In [19] the authors summarize some of the results of large-scale simulations and theoretical approaches about the effects of noise and dimensionality on the scaling behavior of such systems. In [20], the authors introduce a generic phenomenological model for the collective motion of bacteria on a solid agar surface taking into account nutrient diffusion, reproduction, and sporulation of bacteria, extracellular slime deposition, chemo-regulation, and inhomogeneous population. The model is based on a ferromagnetic-like coupling of the velocities of self-propelled particles and is capable of describing the hydrodynamics on the intermediate level. In [21] the authors demonstrate that a system of self-propelled particles exhibits spontaneous symmetry breaking and self-organization in one dimension. They derived a new continuum theory that can account for the development of the symmetry broken state. The collective motion of organisms in the presence of fluctuations is discussed in [22]. In this study Vicsek utilized the simple rule of motion of particles as in [17]. The author demonstrated that there is a transition from disordered to ordered motion at the finite noise level and particles segregate into lanes or jam into a crystalline structure in a model of pedestrians.

A discrete model consisting of self-propelled particles that obey simple interaction rules is studied in [23]. The authors showed that the model can self-organize and exhibit coherent localized solutions in one-dimensional and in two-dimensional spaces. Furthermore, they develop a continuum version of their discrete model and show the agreement between these models.

In [24], Savkin gives a qualitative analysis of the dynamics of a system of several mobile robots coordinating their motion using simple local nearest neighbor rules referring to Vicsek's model in [17]. The author states that under some assumptions the headings of all robots will be eventually the same. Similar analysis was performed by Jadbabaie *et al.* in [25], where they consider both discrete and continuous models as well as leaderless and leader-based situations and show that under certain connectivity conditions the heading of all the agents will converge to the same value, thus providing in a sense a theoretical explanation to the results obtained by Vicsek *et al.* (*i.e.* they considered the model by Vicsek without the additive noise and the position dynamics). Later these results were extended by Moreau [26] and independently by Ren and Beard [27] to more general classes of systems.

Initial studies on flocking from a control theoretic perspective were performed by Tanner and coworkers in [28, 29] using point mass and in [30] considering non-holonomic agents with continuous time dynamics. On the other hand, in a recent study [31] Olfati-Saber developed a theoretical framework for analysis and design of flocking systems with agents with point mass dynamics. He considered two different types of flocking algorithms (which incorporate Reynolds rules): free flocking, in which the agents try to move to a particular distance from its nearest neighbors and also to stay aligned to them, and constrained flocking in which the agents are following virtual agents while performing free flocking. The second algorithm is in principle centralized although it can also be implemented in a decentralized fashion if all the virtual agents for all the individuals have the same dynamics and exchange information initially.

The flocking behavior of multi-agent systems are modeled to work synchronously in many of the studies. On the other hand there have been some studies [32–36] on the asynchronous modeling of multi-agent systems. The work in [32] considers the asynchronous convergence of a linear swarm to a synchronously achievable configuration in the reconfiguration of patterns problems. In this study, a sufficient condition for the asynchronous convergence of a linear swarm to a synchronously achievable configuration is proven to exist. In [33–35] the stability of one-dimensional and M-dimensional asynchronous swarms are studied. In [36] Beni shows that asynchronous swarms may converge in cases in which synchronous swarms may not and that achieving an order from disordered actions is a basic characteristic of swarms and states that "swarms may undergo a transition from non-convergence to convergence as their degree of partial synchronicity diminishes". A study on the aggregation problem is performed in [37] with agents that are anonymous, homogeneous, memoryless, and lack communication capabilities. In a similar study in [38] the authors showed that asynchronous autonomous agents which have limited visibility and no memory, would gather at the same location in finite time provided that they have a compass.

Aggregation in biological swarms were initially modeled and simulated by biologists [9–12]. Inspired by these works, a recent series of studies [33–35, 39–43] has provided rigorous stability and convergence analysis of swarm aggregations based on artificial potential functions both with continuous-time and discrete-time formulations. Particularly, in [39, 40] a biologically inspired n-dimensional (where n is arbitrary) continuous time synchronous swarm model based on artificial potentials is considered and some results on cohesive swarm aggregation have been obtained. Similar results based on artificial potentials and virtual leaders have been independently obtained by Leonard and coworkers in [44, 45] for agents with point mass dynamics. The papers [33–35] focus on asynchronous swarm models with time delays for swarm aggregation in discrete-time settings. In [43], which has more emphasis on design than analysis a particular control strategy for aggregation in swarms has been developed based on artificial potential functions and sliding mode control, assuming simple integrator agent dynamics with model uncertainties and disturbances.

Later in [46] the results in [43] were extended to a significantly more realistic and more difficult setting with non-holonomic unicycle agent dynamics models, again using the tools of artificial potential functions and sliding mode control, but in a slightly different way than [43]. Furthermore, in [47] the results were further extended to include the foraging and formation control problems (in addition to the aggregation problem considered in [46]). One very recent survey that considers multi-agent systems from the perspective of control engineering can be found in [48].

Since in nature and in robotic applications the autonomous agents mostly act asynchronously, a model based on asynchronous actions of agents would be more realistic and implementable. Hence, in this study we will develop an asynchronous version of the model developed by Vicsek [17] (without the additive noise) and investigate the effects of asynchronism in the coordination of agents striving to travel with a common heading. We consider 3 different orientation rules (rules of dynamics to achieve a common heading) and compare the behavior of the self-propelled particle systems for these three different rules. Furthermore, we consider the effect of restricting the maximum turning angle of the particles and perform simulations for bounded and unbounded regions. We perform extensive simulations with many different initial conditions. Moreover, in the discussions section we provide some analytical explanations for the obtained simulation results.

Recently, there have been some articles on extending the works in [25–27] to systems with time delays or systems operating asynchronously [49–53]. In [49], Angeli and Bliman provide an extension of the result by Moreau [26] by relaxing the convexity assumption and allowing for a known and bounded time-delay. The article in [51], besides discussing some available results in the literature, presents some new results for systems/protocols with delays as well. Asynchronous motion is not considered in [49, 51]. The article in [50] summarizes the recent results on synchronous consensus protocols, briefly discusses asynchronous protocols, poses some open questions, and shows some simulation based preliminary results on asynchronous protocols using a custom Java based simulator. The more recent article in [52] presents some new results on the asynchronous agreement problem. No time delays are considered in [52]. The article in [53] considers the problem of asynchronous agreement of systems also incurring time delays and extends the results in [25–27].

Despite the fact that almost all of the above articles claim that they consider the Vicsek's model, in reality they consider only part of the dynamics of the model considered by Vicsek. The model considered is, in general, a linear averaging (or sometimes nonlinear convex/contracting) agreement model that does not include the agent position dynamics which are present in the Vicsek's model. Then the articles investigate the agreement properties in the dynamics of that partial model under some artificial connectivity assumptions. However, in the model by Vicsek the connectivity is an *emergent property* which depends also on the position dynamics of the agents. As a difference from the studies in [24–27, 49–53], in this article we include also the position dynamics of the agents and study the performance of the system for three different agreement strategies for synchronous and asynchronous cases incurring also time delays. In addition, we impose also turn angle restrictions (a type of non-holonomic constraint) on the agents and investigate the performance for different levels of restrictions. For comparing the performances of the strategies we define several performance metrics, which include not only orientation agreement of the agents but also their clustering performance. To the best of our knowledge, no study similar to this one has been performed so far in the literature.

II. HIGH LEVEL DYNAMICS

We consider a multi-agent system consisting of n so called self-propelled or self-driven particles each of which, similar to the model by Vicsek [17], moves based on the dynamics

$$x_i(t+1) = x_i(t) + v\cos(\theta_i(t+1)) \quad (1)$$
$$y_i(t+1) = y_i(t) + v\sin(\theta_i(t+1))$$
$$i = 1, \ldots, n \quad (2)$$

where $x_i(t), y_i(t) \in \mathbb{R}$ denote the cartesian position coordinates of agent i and $\theta_i(t) \in \mathbb{R}$ denotes its orientation angle at time t. We assume that v is constant and equal for all agents. In other words, we assume that all the agents move with the same constant speed in possibly different directions (determined by their orientation angles θ_i). Moreover, we assume that an agent has limited sensing capabilities and can "see" or "sense" the other agents that are within a circle of radius δ from it and call these agents its neighbors. Furthermore, it is assumed that the agents update their orientation based on its current orientation and the orientation of its current neighbors. In particular, we will utilize three different orientation rules, with which the agents will adjust their headings.

Many studies in the literature assume that the agents move synchronously and have perfect information about the orientations of their neighbors. In other words, it is assumed that the agents move simultaneously/synchronously and at each step they know the current positions/orientations of their neighbors. However, in real swarms this is hardly possible. Implementing such dynamics will require a global clock to be shared by all the agents. Therefore, an asynchronous model is more realistic in which each agent can move and reorient itself independently. Moreover, usually there might be time delays in the communication/sensing between the agents. Including such delays in the multi-agent systems will result in more realistic approaches. In order to achieve such a realistic model we use a higher-level asynchronous model similar to the one used in [54].

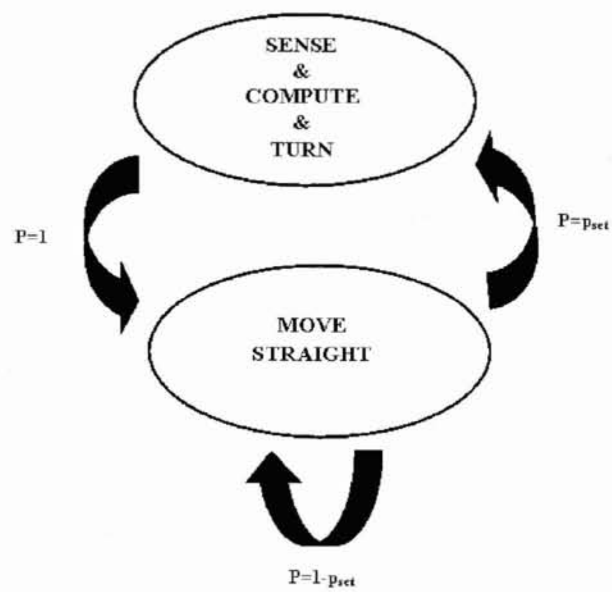


Fig. 1. Finite state machine model.

For the asynchronous high level model we will refer to the study in [55] which considers the cyclic pursuit problem with asynchronous high level dynamics. In that study a finite state machine (FSM) is proposed for high level model. The architecture consists of three behaviors: *wait*, *sense and compute*, and *move*. In our model here the agents always move in the last updated direction with constant speed and a finite state machine works for the orientation dynamics. Therefore, the behavior considered here can be described with the *sense-compute-turn* and *move straight* states (Fig. 1).

During the *sense-compute-turn* behavior the i^{th} agent gets (measures or receives by other means) the orientations of neighbor agents and computes its own next desired orientation and turns to the computed orientation. During the *move straight* behavior, the agent doesn't turn, that is, basically moves along its last updated orientation. These behaviors are arbitrated by using a finite state machine, in an infinite loop as shown in Fig. 1. As soon as the *sense-compute-turn* state is completed agents immediately pass to the *move straight* state (with the probability of 1). However, the *move straight* state is followed by the *sense-compute-turn* state depending on some probabilistic measure or reasoning. Let the following state of *move straight* state be *sense-compute-turn* state with probability of p_{sct} and again *move straight* state with probability of $1 - p_{sct}$.

We assume that each agent has a low level control which guarantees that the agent turns to the computed orientation with the specified angular speed. We are not concerned with the low level dynamics and how the low-level control is implemented. Therefore, the analysis below is applicable for many systems with variety of different low-level vehicle dynamics including heterogenous swarms/systems (*i.e.* swarms consisting of more than one type of agents). Moreover, in the current study we ignore the issue of collisions between the agents. The resulting sequence of behaviors can be summarized as: turn to the computed orientation. Wait for a predetermined time interval. Then sense the orientations of the neighbor agents and turn again in the computed orientation.

Recall that during the *sense-compute-turn* behavior the i^{th} agent gets the orientations of the neighbor agents and then computes its next desired orientation. However, during these sensing and computing processes of the i^{th} agent the neighbor agents may be in their *sense-compute-turn* state and therefore the measured orientations of the neighbor agents may be outdated. Moreover, the measurement of the orientations of the neighbor agents may itself incur some delays. Whether any type of sensors or even communication are used, the propagation delay of the signals may lead to measurement of old (outdated) orientations. Similarly, delay will also be present even if the orientations are obtained by inter-agent communication. Therefore, the modeling of the dynamics of agents working for a common orientation problem should be designed taking into account the orientation sensing delays. Referring to this phenomena we introduce the variables $\tau^i_j(t)$ which satisfy $0 \leq \tau^i_j(t) \leq t$ in order to represent the delay in the orientation measurements. In other words, we assume that at time t agent i knows $\theta_j(\tau^i_j(t))$ instead of the actual $\theta_j(t)$ about the orientation of agent j. In other words, $\theta_j(\tau^i_j(t))$ is the *perceived orientation* of agent j by agent i at time t. Also, since each agent operates on its local clock following the state machine cycle on Fig. 1 without a need for synchronization with the other agents, we introduce a set of time indices T^i, $i = 1, 2, \ldots, n$, at which the agent i updates its orientation θ_i where the sets T^i are independent subsets of the set $\{0, 1, 2, \ldots\}$. It is assumed that at the other instances the agent i does not perform orientation calculation (it might be in one of the other states/behaviors at these time instants). Note that in the synchronous model $\tau^i_j(t) = t$ and $T^i = \{0, 1, 2, \ldots\}$ for all $t > 0$ and $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, n$. In other words, in the synchronous case all the agents have the exact and current orientation information of their neighbors and perform updates at all time instants simultaneously/synchronously.

We believe that the asynchronous model is more realistic (compared to the studies performed earlier with a synchronism assumption) since in real world applications (such as robots coordinating to achieve a common orientation) or animal flocks (such as schooling behavior of fish) usually there is no synchrony between agents and time delays are also possible. In our model we utilize the study on the relation between the synchronism and asynchronism in the parallel computing systems in [56].

The asynchronism between the agents (*i.e.* robots, fish) may be at different levels due to the characteristics of each agent itself or some environmental disturbance. In some multi-agent systems the asynchrony may be negligible (leading to a synchronism assumption) and the behavior of these systems may be computable or predictable. On the other hand, in some systems asynchronism may drastically change the performance of the system. In these kind of systems the behaviors of agents may be difficult to predict. Nevertheless, unbounded or excessively long delays in the information flow or acting of the agents may result in the violation of "agent interaction" concept of multi-agent systems. In other words, it might be difficult to view the systems experiencing unbounded or excessively long delays or systems with agents some of which do not act for unbounded amount of time as a single multi-agent system. This is because an agent that never performs sensing or never acts cannot be considered as a member of the group. Therefore, the level of asynchronism should be limited in such a way that the agents still can interact and form a single multi-agent system. Hence, here we state an assumption (as utilized in the study [55]) which establishes a bound on the maximum possible time delay as well as guarantees uniformity in the updates of the agents.

Assumption 1. There exists a positive integer B such that

(a) For every i and every $t \geq 0$, at least one of the elements of the set $\{t, t+1, \ldots, t+B-1\}$ belongs to T^i.
(b) There holds $t - B < \tau^i_j(t) \leq t \ \forall i, \ t \geq 0, t \in T^i$.

This assumption basically states that (i) every agent performs an update or change in its orientation in at most B time steps; (ii) the delay in sensing the orientations of the neighbors of the agent is bounded as well by at most B time steps. This assumption in a sense restricts the level of asynchronism in the multi-agent system. Assumption 1 is taken from [56] where it is utilized for parallel and distributed computing systems. Systems satisfying Assumption 1 are being referred to as partially asynchronous systems in the parallel and distribution computation literature [56].

Let

$$N_i(t) = \{j : |j \neq i, (x_i(t) - x_j(t))^2 + (y_i(t) - y_j(t))^2 \leq \delta^2\}$$

denote the actual set of neighbors of agent i at time t and $|N_i(t)|$ denote the number of agents in the set $N_i(t)$.

For the asynchronous system which also considers the time delays in sensing/communication the set of neighbors of agent i at time t can be described as

$$\mathbb{N}_i(t) = \{j : |j \neq i, (x_i(t) - x_j(\tau^i_j(t)))^2 + (y_i(t) - y_j(\tau^i_j(t)))^2 \leq \delta^2\}$$

where $\tau^i_j(t)$ represents the last instant at which agent i obtained the orientation information of agent j during the last *sense-compute-turn* state. Note that during the delays in the information gathering and computing states the neighbor agent j may leave the neighborhood region or any other agent may enter this region.

Below we describe three different strategies for the orientation computation of the agents.

III. STRATEGIES FOR ORIENTATION AGREEMENT

3.1 Strategy 1 (averaging)

This strategy is based on the averaging of orientations of neighbor agents. The new orientation of agent i at step $t+1$ is determined by the following equation:

$$\theta_i(t+1) = \frac{1}{1+|\mathbb{N}_i(t)|}\left(\theta_i(t) + \sum_{j\in\mathbb{N}_i(t)} \theta_j(\tau^i_j(t))\right), \quad t \in T^i \tag{3a}$$

$$\theta_i(t+1) = \theta_i(t), \quad t \notin T^i. \tag{3b}$$

As mentioned above, we assume that at time t agent i knows $\theta_j(\tau^i_j(t))$ instead of the actual $\theta_j(t)$ about the orientation of agent j. In other words, $\theta_j(\tau^i_j(t))$ is the *perceived orientation* of agent j by agent i at time t. Consequently, if agent i has not yet obtained any information about the j^{th} agent's orientation and still has the initial orientation information, then $\tau^i_j(t) = 0$ whereas $\tau^i_j(t) = t$ means that agent i has the current orientation information of the j^{th} agent. The difference between the current time t and the value of the variable $\tau^i_j(t)$, *i.e.*, $(t - \tau^i_j(t))$ is the delay occurring due to the sensory, computing and/or communication processes or other reasons. Note from Assumption 1 that $t - \tau^i_j(t) < B$ should be satisfied.

One drawback with the rule in (3a) is that it may sometimes result in directions of motion that are not very intuitive. For example, assume that there are two agents with directions of motion $+5°$ and $+355°$. Based on the rule in (3a) on the next step both the agents will turn to $180°$ (*i.e.* they will flip direction) while the intuitive direction is $0°$ for both. Here we assumed that the orientation angles are defined between $0°$ and $360°$. The situation will not change if they are defined between $-180°$ and $+180°$ since the same problem will occur around $180°$ this time. The reasons for such behavior are discussed in more detail in the discussion section.

3.2 Strategy 2 (relative angles)

In this strategy the agents determine their new orientations by considering the orientation differences between themselves and their neighbors or basically the relative orientations. The next orientation of agents is found by the following equation:

$$\theta_i(t+1) = \theta_i(t) + \frac{\Theta}{1+|\mathbb{N}_i(t)|}, \quad t \in T^i \tag{4a}$$

$$\theta_i(t+1) = \theta_i(t), \quad t \notin T^i \tag{4b}$$

where Θ is the total of differences between the orientation of the agent itself and the orientations of its neighbor agents. The orientation of an agent itself and its neighbor's orientation may appear in four different arrangements (Fig. 2). Therefore, Θ is calculated considering the four different cases with the following pseudo code.

```
Θ = 0
FOR j ∈ ℕ_i(t)
    Δ = θ_j(τ^i_j(t)) − θ_i(t)
    IF Δ ≤ π OR Δ ≥ −π   (CASE 1 OR 3)
      Θ = Θ + Δ
    ELSE IF Δ>π   (CASE 2)
      Θ = Θ + Δ − 2π
    ELSE IF Δ<−π   (CASE 4)
      Θ = Θ + Δ + 2π
    END
END
```

where $\theta_i(t) \in [0, 2\pi)$ $\forall i, t$. This pseudo-code is also equivalent to the equation

$$\Theta = \sum_{j\in\mathbb{N}_i(t)} mod(\theta_j(t) - \theta_i(t) + \pi, 2\pi) - \pi. \tag{5}$$

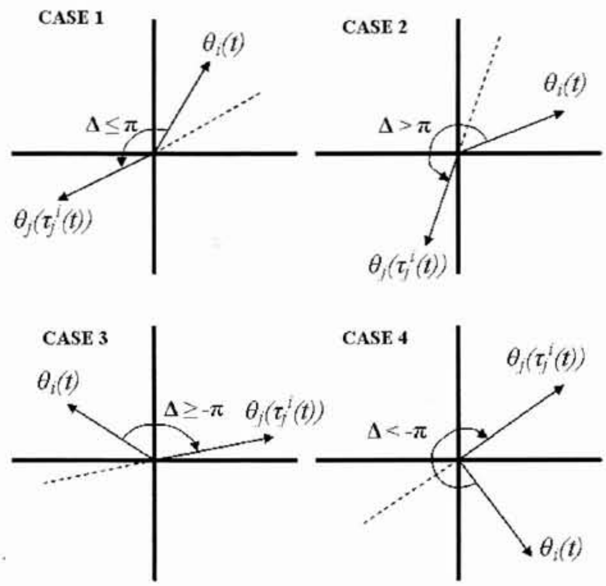


Fig. 2. Possible arrangements of orientation of an agent and its neighbor's orientation.

The rule in (4a) is more intuitive compared to rule (3a) since it considers the relative orientations and always chooses the smaller angle. However, it may also have problems in the cases of some symmetries. For example if three agents are oriented such that the orientation difference between each pair is 120°, they will continue their motion with their previous orientations and will not achieve orientation agreement. The details of such behavior are also discussed in the discussions section.

3.3 Strategy 3 (vector sum)

Strategy 3 is based on the vectorial sum of unit vectors that lie along the orientations of agents. Let r_i denote the unit vector in the direction of motion of agent i. Then the orientation rule becomes

$$\theta_i(t+1) = angle\left(r_i(t) + \sum_{j\in\mathbb{N}_i(\tau_j(t))} r_j(\tau_j^i(t))\right), \quad t \in T^i \tag{6a}$$

$$\theta_i(t+1) = \theta_i(t), \quad t \notin T^i \tag{6b}$$

where $angle(\upsilon)$ is the function that returns the angle of any given vector υ. In implementations it can be computed by using the $atan2(Py, Px)$ function where Px and Py are the components of the computed vector on the right hand side of (6a) along the x and y directions, respectively.

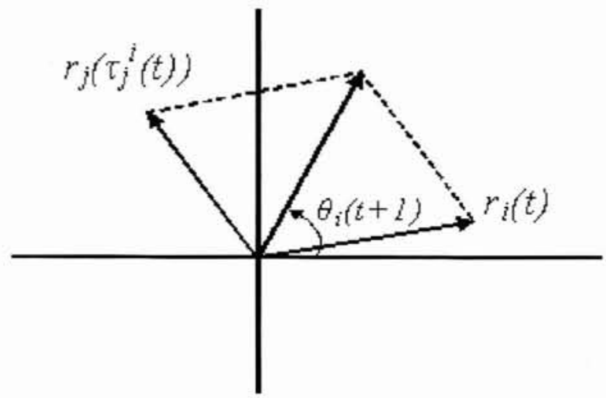


Fig. 3. Orientation rule of Strategy 3 for only one neighbor (j) of agent i.

For instance in Fig. 3 we see the calculation of new orientation, $\theta_i(t+1)$ of i^{th} agent with only one neighbor (j).

Rule (6a) is another intuitive rule that is used in determining the agent directions. However, as was the case with rule (4a) it may be difficult to decide the next orientation using rule (6a) in some cases with symmetry. However, these cases have very low probability in a swarm of many agents. In implementation if such a case occurs one may choose the turning direction randomly until the symmetry is broken. The limitations of Strategy 2 and 3 in cases of symmetries and the drawback of Strategy 1 explained in Section 3.1 are investigated in more detail in the discussion section.

IV. TURN ANGLE RESTRICTIONS

As discussed previously we assume that the agents update their orientation based on their own orientation and the orientation of their neighbors. In this section we additionally assume that there is a restriction on the maximum possible turning angle of the agents due to mechanical or physical reasons. Therefore, the dynamics of the orientation angles of the agents are given by

$$\theta_i(t+1) = \theta_i(t) + min(abs(\phi_i(t)), \alpha) \times sign(\phi_i(t)), \quad t \in T^i \tag{7a}$$

$$\theta_i(t+1) = \theta_i(t), \quad t \notin T^i \tag{7b}$$

where α is the maximum possible turn angle per step and $\phi_i(t)$ is the desired turn angle which is computed at time t by using one of the three strategies given above. Due to (7a), during an update i^{th} agent can turn at most the angle α in the direction of $\phi_i(t)$ (clockwise or counter clockwise).

Adding the turn angle restrictions we can rearrange the rules of the previous three strategies for computing $\phi_i(t)$ as

For Strategy 1

$$\phi_i(t) = \frac{1}{1+|\mathbb{N}_i(t)|}\left(\theta_i(t) + \sum_{j\in\mathbb{N}_i(t)} \theta_j(\tau_j^i(t))\right) - \theta_i(t), \quad t \in T^i. \tag{8}$$

For Strategy 2

$$\phi_i(t) = \frac{\Theta}{1+|\mathbb{N}_i(t)|} \quad t \in T^i \tag{9}$$

where Θ is calculated by the pseudo code given in Section 3.2.

For Strategy 3

$$\phi_i(t) = angle\left(r_i(t) + \sum_{j\in\mathbb{N}_i(\tau_j(t))} r_j(\tau_j^i(t))\right) - \theta_i(t), \quad t \in T^i. \tag{10}$$

We would like to emphasize here that having turn angle restrictions is a very realistic assumption since most real agents will have such constraints. However, such restrictions have not been considered in the literature so far.

Table 1. Pseudocode.

```
Draw the initial positions and orientations of agents
randomly from a uniform distribution
FOR i = 1:n DO
    set Ω_i = 1; (Ω_i is the number of steps
        that agent i has not performed an update.
        At the beginning Ω_i is one for every agent.)
END
FOR t = 1:T DO (T = number of simulation steps)
  FOR i = 1:n (n = number of agents) DO
    prbOfUpdate = 1/(B-Ω_i + 1);
    c = randint(1, 100); (Generate random integer from
        uniform distribution to compare with prbOfUpdate)
    IF (c ≤ prbOfUpdate * 100) (performing update)
      FOR j = 1:n DO
        τ = randint(t-B,t)
          (τ is the random step drawn between last
          step orientation updated and current step
          Consider τ as the last step that agent i
          received/measured the orientation of agent j)
        IF agent j is neighbor of agent i
          Use θ_j(τ) in the computation of the new
          orientation
          (based on formula for the current strategy)
        END
      END
      Ω_i = 1; Set to one since the agent is updated
    ELSE (No update will be performed)
        Ω_i = Ω_i + 1; (increment number of steps
          that no update performed)
        No change in agent i's orientation
    END
  END
END
```

V. NUMERICAL SIMULATION RESULTS

We simulated the motion of $n = 50$ agents. Initially the agents are located in a square region of size 100×100 units and the constant speed of all of the agents is set to 1 unit/step. The simulations are performed for T time steps ($T = 500$ for unbounded region and $T = 1000$ for bounded region). The agents perform updates depending on a stochastic function of B. The probability of update of an agent is distributed along the interval $t \in (t-B, t]$ such that it is $1/(B-\Omega+1)$ where Ω is the number of steps that the agent has not performed an update since the last update. In other words, if the agent has just performed an update in the previous step the probability that it will perform an update again is $1/B$, whereas if it has not performed an update for $3<B$ steps then its probability of update is $1/(B-2)$. If the agent has not performed an update for B time steps then its probability of update becomes 1. At each time step at which the agent does not perform an update the value of Ω is incremented by 1 and at each time step the agent performs an update, the value of Ω is reset to zero. Note that this implementation is not a real discrete event based asynchronous system. Instead it mimics such systems and is sufficient for illustrating/verifying the performance of the asynchronous system discussed in this article. The initial positions and orientations of agents are generated randomly and for each simulation the same initial conditions are utilized. We provide 20 simulation results performed for different initial conditions generated randomly and present the mean and variance of the results. The algorithm performing these steps is presented in Table 1. In order to measure the performance of the system we used the following four performance metrics

$$e_d(t) = \frac{2}{n(n-1)}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\|z_i(t)-z_j(t)\|, \quad t \geq 1$$

$$e_\theta(t) = \frac{2}{n(n-1)} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \|\theta_i(t) - \theta_j(t)\|, \quad t \geq 1$$

$$\dot{e}_\theta(t) = \frac{1}{n} \sum_{i=1}^{n} \|\theta_i(t) - \theta_i(t-1)\|, \quad t \geq 2$$

where $z_i(t) = [x_i(t), y_i(t)]^T$ is the position vector of agent i. Basically $e_d(t)$ is the *average distance between the agents*, and $e_\theta(t)$ is the *average of orientation differences between the agents*, and $\dot{e}_\theta(t)$ is the *average rate of change of orientation of the agents* at time t. The fourth performance metric is the number of clusters formed by the agents. A cluster is defined as the group of agents which are connected to (meaning "are neighbors of") each other either directly or indirectly through other agent. (Note that agents i and j belong to the same cluster at time t if $\|z_i(t) - z_j(t)\| \leq \delta$.) As $e_d(t)$ gets lower, the agents get closer to each other which implies that the multi-agent system performs better in clustering. The metric, number of clusters, is also a performance criterion in determining the success of clustering of agents. Note that as the number of clusters decrease or size of clusters increase (resulting in lower $e_d(t)$) the number of agents traveling with the same heading increases. Therefore, if the performance in clustering is better, then the performance in orientation agreement is also better. On the other hand, as $e_\theta(t)$ gets lower, the agents are heading in closer orientations which means they have better performance in orientation agreement. The agents have more steady headings if $\dot{e}_\theta(t)$ converges to zero.

We have conducted simulations in order to determine the effects of asynchronism and time delays on the cluster formation by varying α (turn angle restrictions) in unbounded and bounded regions.

5.1 Effect of α (unbounded region)

Here, we compare the effects of α and asynchronism on the performances of the three orientation strategies. δ is set to 20 units. Note that in all simulations $B = 0$ corresponds to the synchronous case and $B = 10$ corresponds to an asynchronous case.

5.1.1 Effect of α for unbounded region in synchronous model

In Figs. 4 and 5 we plotted $e_d(t)$ and $e_\theta(t)$ at the end of simulations ($t = T = 500$) versus α values. It is seen that as the α value increases (which means that the restriction on the turning angle decreases) the values of both $e_d(T)$ and $e_\theta(T)$ decrease in all strategies. The total distance between agents is smaller for higher

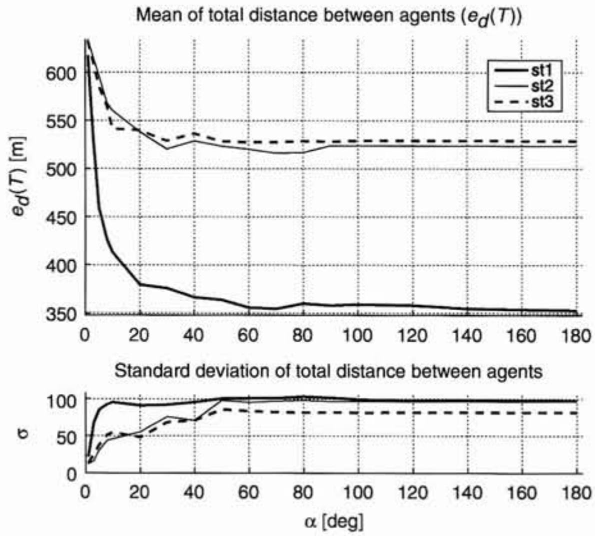


Fig. 4. $e_d(T)$ for strategies 1 (bold solid line), 2 (solid line), 3 (dash-dot line) for synchronous case and unbounded region.

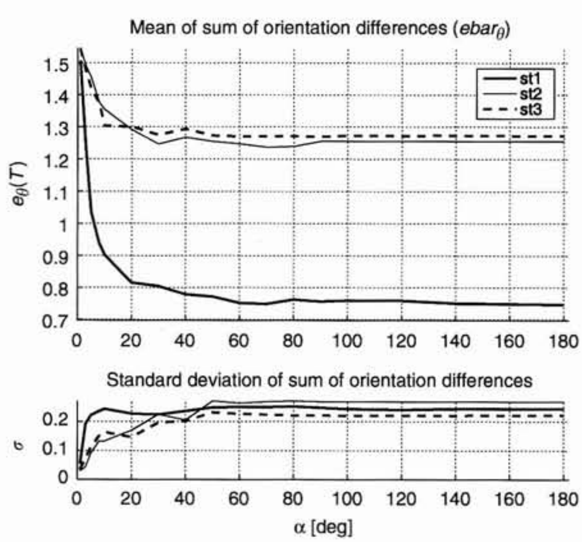


Fig. 5. $e_\theta(T)$ for strategies 1 (bold solid line), 2 (solid line), 3 (dash-dot line) for synchronous case and unbounded region.

α values. This is an expected result when we consider the fact that as the restriction gets weaker, the agents perform better turning motion and therefore, they aggregate better. On the other hand, the decrease in the sum of orientation differences $e_\theta(T)$ implies that the number of agents moving with different headings decrease and also the difference between the heading of clusters decreases, since the orientation adjustment becomes easier at lower turning restrictions. In Fig. 6 we see the clustering performance of the three strategies with varying α. As seen all strategies get better as the turning angle restrictions get smaller.

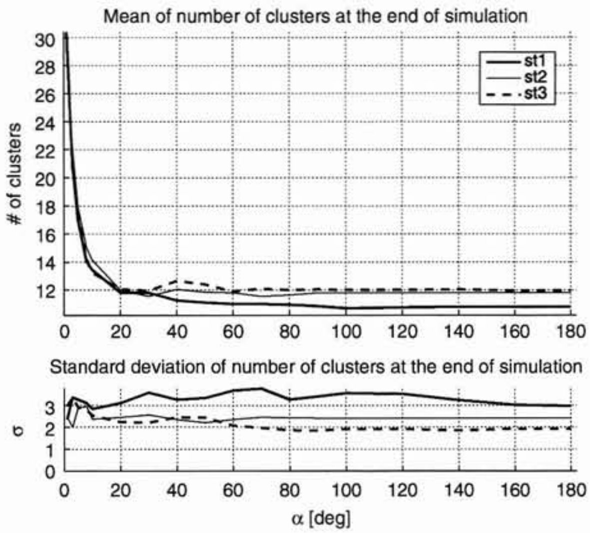


Fig. 6. The number of clusters at $t = T$ for strategies 1 (bold solid line), 2 (solid line), 3 (dash-dot line) for synchronous case and unbounded region.

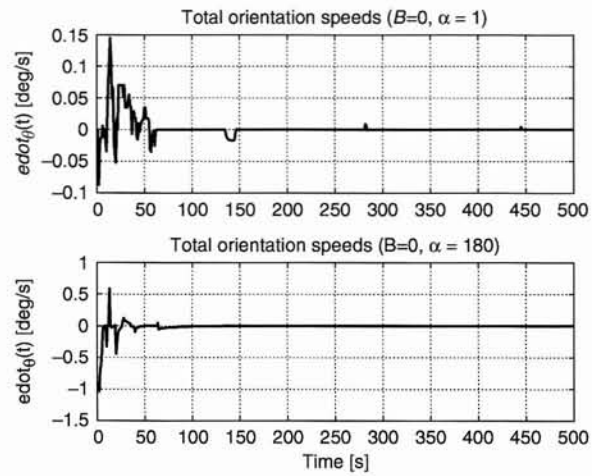


Fig. 7. Total of rate of change of orientations of agents at each step for $\alpha = 1°$ (upper subplot) and $\alpha = 180°$ (lower subplot) for an arbitrary initial condition (Strategy 1 - synchronous case - unbounded region).

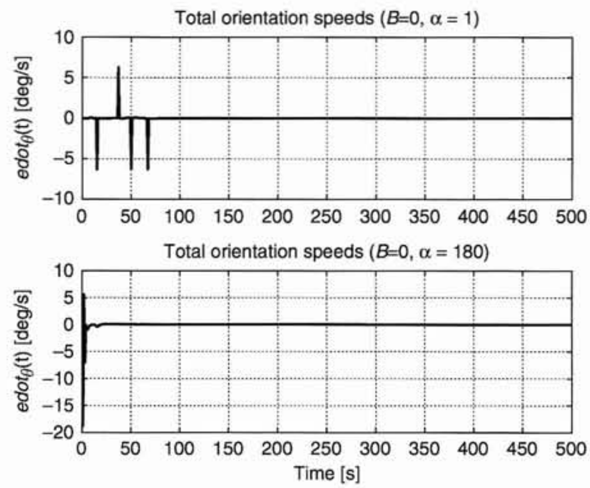


Fig. 8. Total of rate of change of orientations of agents at each step for $\alpha = 1°$ (upper subplot) and $\alpha = 180°$ (lower subplot) for an arbitrary initial condition (Strategy 2 - synchronous case - unbounded region).

In all Figs. 4, 5, and 6 the first strategy has the best performance for all α values. The second and third strategies are close to each other.

The total of rate of change of orientations of all agents at each step, $\dot{e}_\theta(t)$ is plotted in Figs. 7, 8, and 9 for strategies 1, 2, and 3, respectively. In all three figures the results for $\alpha = 1°$ show lower values than the ones for $\alpha = 180°$ because as α gets lower the maximum possible rate of change of orientations of each agent gets lower as well. The agents having lower α values cannot form large clusters as stated above. Therefore, they continue their motion in relatively small clusters (there are many of them) which are spreading away. As the clusters get out of neighborhood range of each other with different orientations then there exits no possibility for them to change their orientations to travel with the same heading. Therefore, in all figures, after some amount of steps $\dot{e}_\theta(t)$ settles to zero implying that no change of orientations of clusters or agents occur. Note that in Fig. 7 for $\alpha = 1°$ there are some peaks around 140^{th}, 270^{th} and 440^{th} steps that means some small clusters come across.

5.1.2 Effect of α for unbounded region in asynchronous model

Here, we present the results of simulations of the three strategies for the asynchronous case. In Figs. 10 and 11 we plotted $e_d(t)$ and $e_\theta(t)$ at the end of simulations ($t = T = 500$) versus α values. Like the results for the synchronous case as the α value increases the values of both $e_d(T)$ and $e_\theta(T)$ decrease in all strategies. Comparing with the synchronous case, we can conclude that in the asynchronous case all strategies have worse performances. This is an expected result when we consider the delays in the sense and computing states (which result in lack of valid information about the orientations of its neighbor agents) of the agents and the fact that the agents may not perform orientation update at each time step. These two reasons make the performances worse. In Fig. 12 we see the clustering performance of the three strategies with varying α for the asynchronous agents.

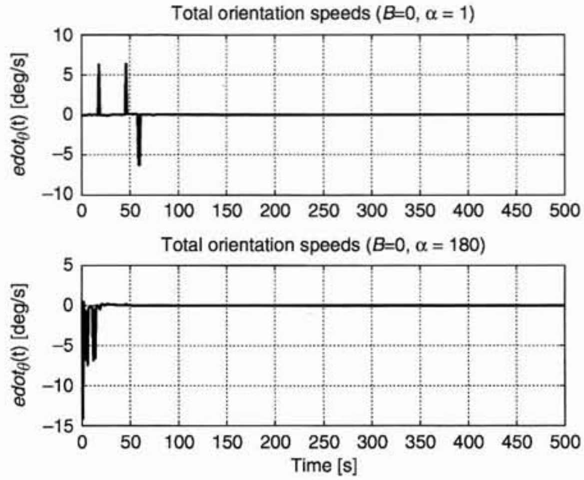


Fig. 9. Total of rate of change of orientations of agents at each step for $\alpha = 1°$ (upper subplot) and $\alpha = 180°$ (lower subplot) for an arbitrary initial condition (Strategy 3 - synchronous case - unbounded region).

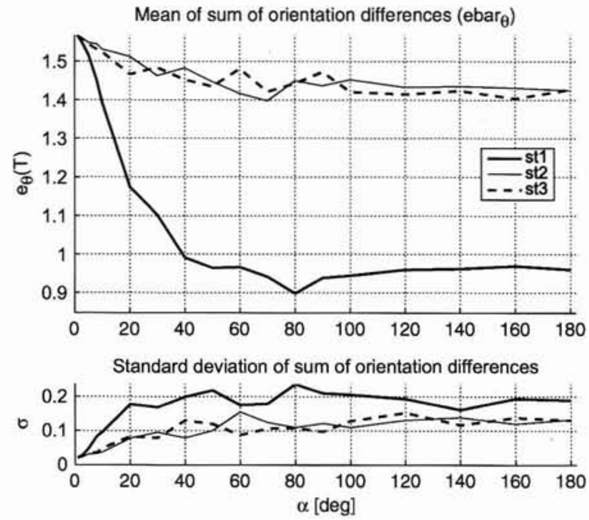


Fig. 11. $e_\theta(T)$ for strategies 1 (bold solid line), 2 (solid line), 3 (dash-dot line) for asynchronous case and unbounded region.

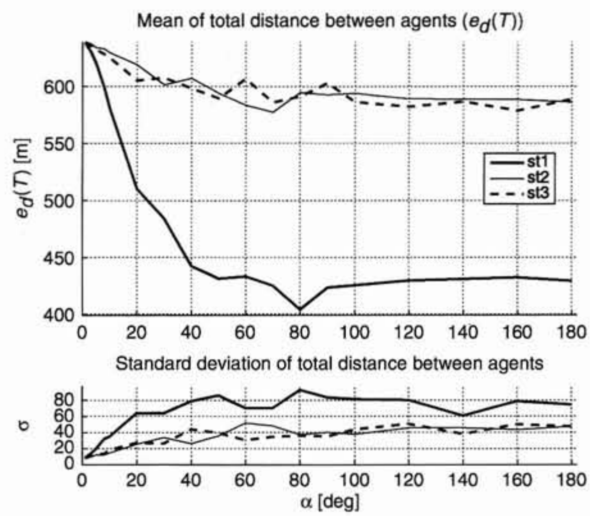


Fig. 10. $e_d(T)$ for strategies 1 (bold solid line), 2 (solid line), 3 (dash-dot line) for asynchronous case and unbounded region.

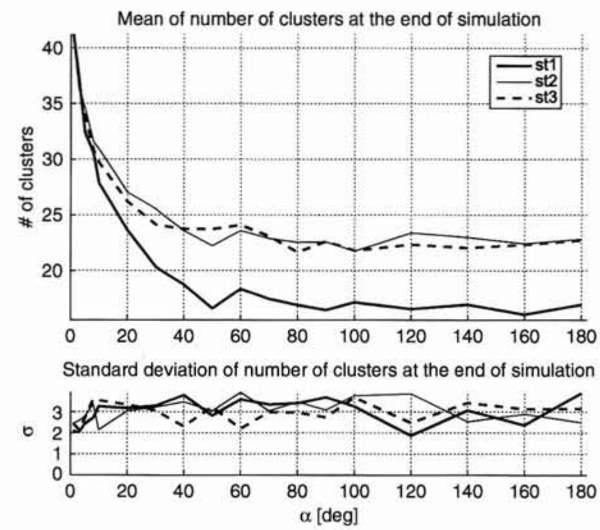


Fig. 12. The number of clusters at $t = T$ for strategies 1 (bold solid line), 2 (solid line), 3 (dash-dot line) for asynchronous case and unbounded region.

As seen all strategies perform better as the turning angle restrictions get weaker. However, like the worse performances in $e_d(t)$ and $e_\theta(t)$ with respect to the synchronous case, the number of clusters for a specific α value is worse (higher) too, for the asynchronous dynamics.

Like the synchronous case all plots in Figs. 10, 11, and 12 show that the first strategy again has the best performance for all α values. The performances of the second and the third strategies are close to each other.

The asynchronous results for $\dot{e}_\theta(t)$ are plotted in Figs. 13, 14, and 15 for Strategies 1, 2, and 3, respectively. As in the synchronous case all three figures show that results for $\alpha = 1°$ have lower values than the ones for $\alpha = 180°$ due to lower maximum possible turning angle values at each step. In all figures the amount of steps that $\dot{e}_\theta(t)$ settles to zero is higher with respect to synchronous case. Again note that since the clusters are spread away as time passes the possibility of orientation adjustment with respect to neighbor clusters diminishes. Therefore, the fluctuation amounts are diminishing in

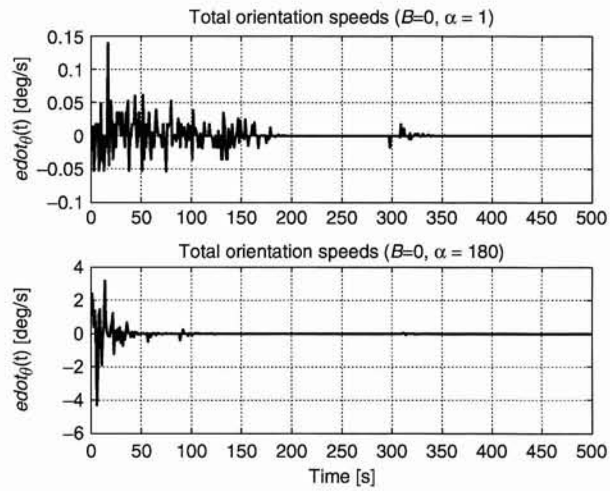


Fig. 13. Total of rate of change of orientations of agents at each step for $\alpha = 1°$ (upper subplot) and $\alpha = 180°$ (lower subplot) for an arbitrary initial condition (Strategy 1 - asynchronous case - unbounded region).

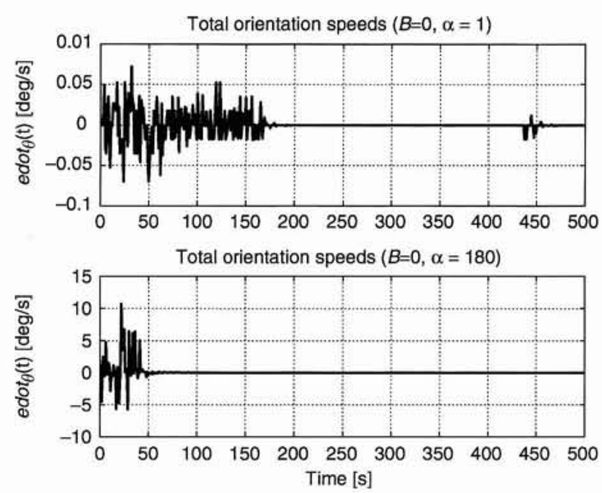


Fig. 14. Total of rate of change of orientations of agents at each step for $\alpha = 1°$ (upper subplot) and $\alpha = 180°$ (lower subplot) for an arbitrary initial condition (Strategy 2 - asynchronous case - unbounded region).

time. There are some out of order peaks at the different steps that may be caused by the asynchronism in the dynamics or some clusters come across with each other.

5.2 Effect of α (bounded region)

The simulation parameters and initial conditions used in this section are the same with those used in the previous sections except that in this case we restrict the arena in which the agents move. They move in a 100 by

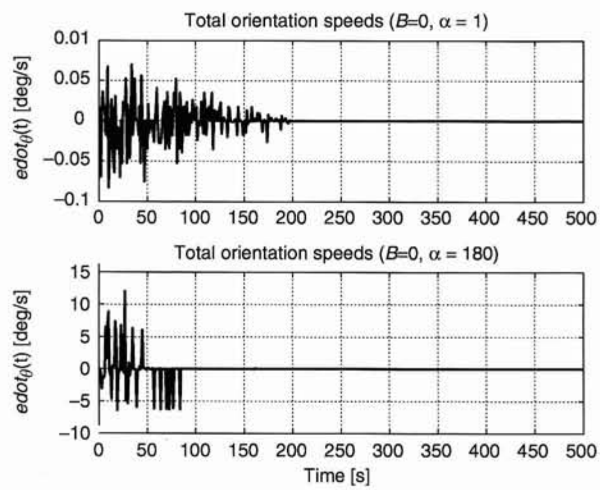


Fig. 15. Total of rate of change of orientations of agents at each step for $\alpha = 1°$ (upper subplot) and $\alpha = 180°$ (lower subplot) for an arbitrary initial condition (Strategy 3 - asynchronous case - unbounded region).

100 square region. When an agent faces any boundary it continues to its motion with the orientation that is the reflection of its previous orientation just like a light beam reflects on a mirror. In the following sections again we present the simulation results for synchronous and asynchronous cases.

5.2.1 Effect of α for bounded region in synchronous model

The plots of $e_d(t)$ and $e_\theta(t)$ are presented in Figs. 16 and 17, respectively. As in the previous results we see that as the α value increases the values of both $e_d(T)$ and $e_\theta(T)$ decrease in all strategies. In Fig. 18 we see the clustering performance of the three strategies with varying α. As seen all strategies get better in clustering as the turning angle restrictions get smaller.

The first strategy has again best performance as seen in Figs. 16, 17, and 18. The second and third strategies perform close to each other. As expected, since the region is bounded the performances of this case are better than performances of its unbounded counterpart.

Figs. 19, 20, and 21 are the plots of the $\dot{e}_\theta(t)$ for synchronous and bounded case. The amounts of rate of changes of the orientations of the agents are very high compared to the unbounded region results. In the unbounded region the agents settle down to some orientation and continue with this orientation for the rest of the simulation. However, if the region is bounded the agents have to turn (to the reflection angle) when they come across with the boundaries. Therefore, we see some sharp peaks in the plots that resulted from

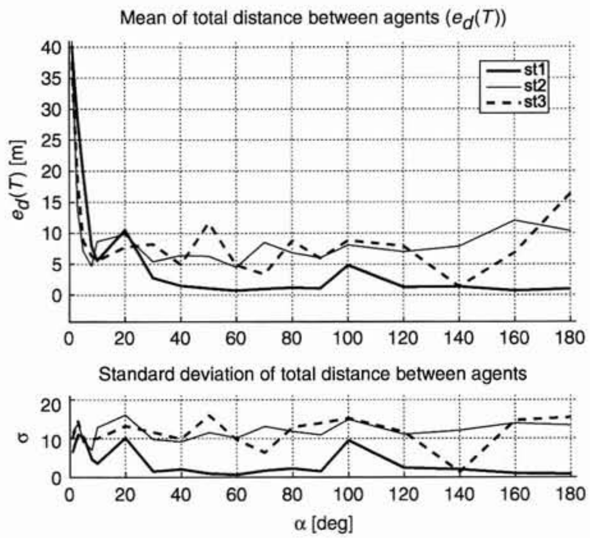


Fig. 16. $e_d(T)$ for strategies 1 (bold solid line), 2 (solid line), 3 (dash-dot line) for synchronous case and bounded region.

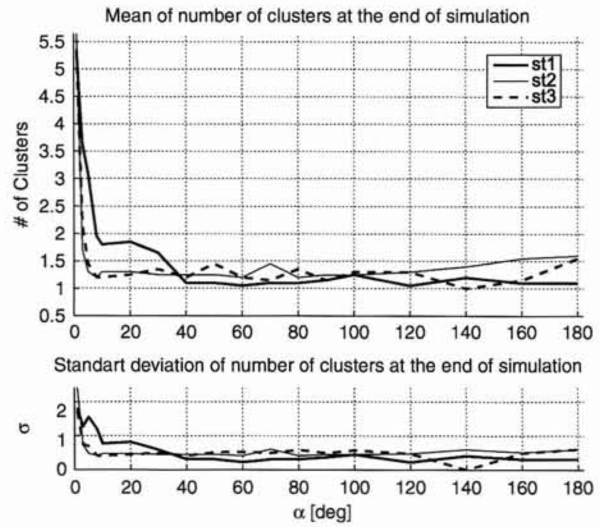


Fig. 18. The number of clusters at $t = T$ for strategies 1 (bold solid line), 2 (solid line), 3 (dash-dot line) for synchronous case and bounded region.

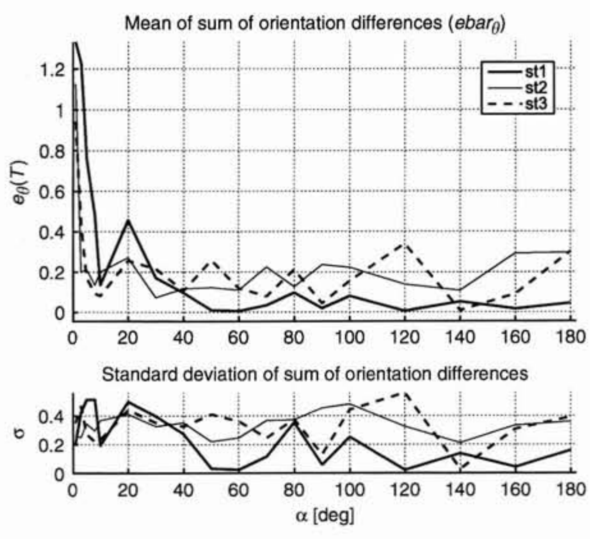


Fig. 17. $e_\theta(T)$ for strategies 1 (bold solid line), 2 (solid line), 3 (dash-dot line) for synchronous case and bounded region.

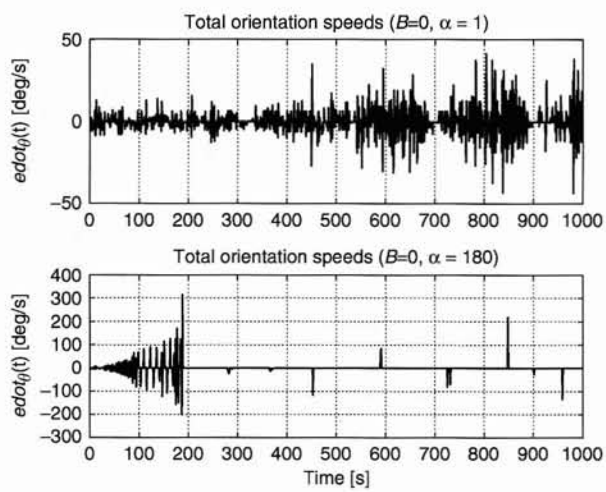


Fig. 19. Total of rate of change of orientations of agents at each step for $\alpha = 1°$ (upper subplot) and $\alpha = 180°$ (lower subplot) for an arbitrary initial condition (Strategy 1 - synchronous case - bounded region).

the reflection of clusters from walls. Note that during this process some of the agents come across with the boundaries before their neighbors that they travel with the same orientations. Since the leader agents turn to a reflection orientation, orientation strategies of all neighbor agents calculate new orientations -different from the one they all settled down. Hence, we see that for the high restriction of turn angles -low α value- the new orientation agreement of a cluster of agents may take more time.

5.2.2 Effect of α for bounded region in asynchronous model

The results of the asynchronous version of the previous bounded region simulation is presented in Figs. 22, 23, and 24. As seen the results are similar in terms of getting better as the turning angle restrictions weaken. Comparing the synchronous and asynchronous results, again we find that the asynchronism leads to worse performance. Here the second and third strategies perform close to each other and worse than the first

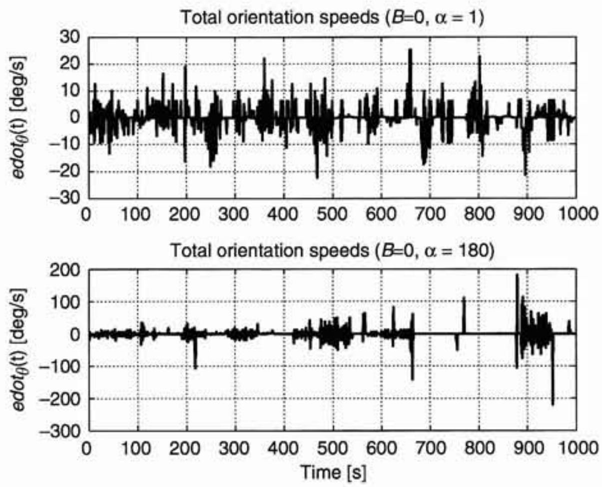


Fig. 20. Total of rate of change of orientations of agents at each step for $\alpha = 1°$ (upper subplot) and $\alpha = 180°$ (lower subplot) for an arbitrary initial condition (Strategy 2 - synchronous case - bounded region).

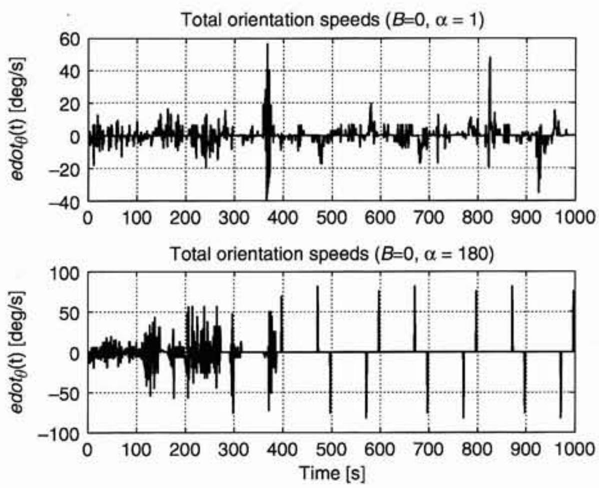


Fig. 21. Total of rate of change of orientations of agents at each step for $\alpha = 1°$ (upper subplot) and $\alpha = 180°$ (lower subplot) for an arbitrary initial condition (Strategy 3 - synchronous case - bounded region).

strategy again. Moreover, as expected since the region is bounded the performances of this case are better than performances of its unbounded counterpart.

The last three Figs. 25, 26, and 27 are the plots of the $\dot{e}_\theta(t)$ for asynchronous and bounded case for strategies 1, 2, and 3, respectively. As seen from the figures for $\alpha = 1$ the agreement of orientations of agents is not achieved at all. Every agent changes its orientation due to its orientation strategy and/or facing of boundaries. The agents having $\alpha = 180°$ perform better than agents $\alpha = 1°$ in this case but they have still worse performance compared to the agents with synchronism assumption.

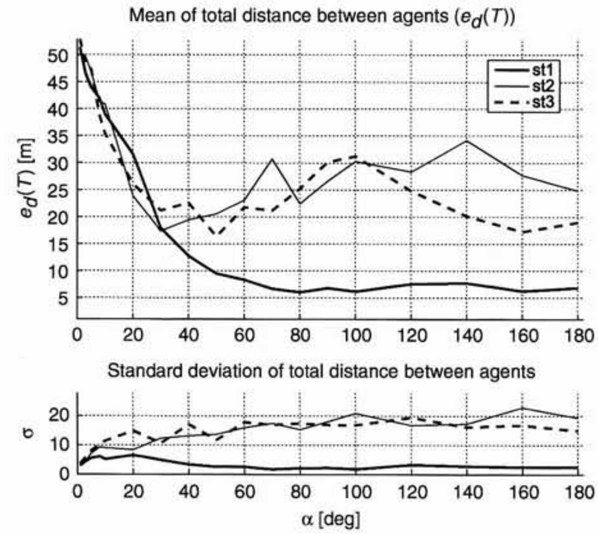


Fig. 22. $e_d(T)$ for strategies 1 (bold solid line), 2 (solid line), 3 (dash-dot line) for asynchronous case and bounded region.

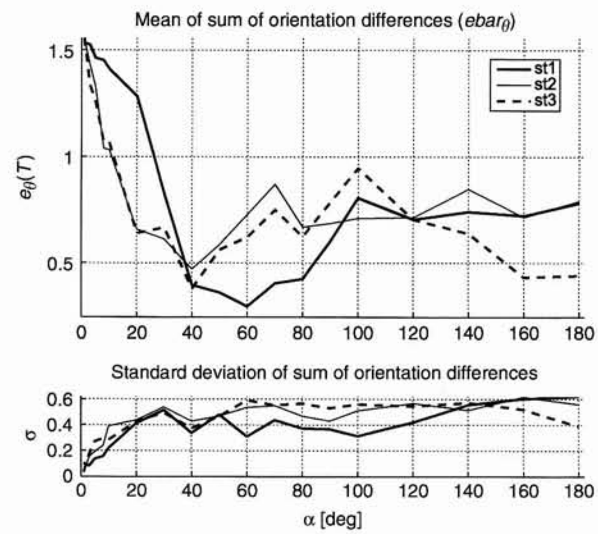


Fig. 23. $e_\theta(T)$ for strategies 1 (bold solid line), 2 (solid line), 3 (dash-dot line) for asynchronous case and bounded region.

VI. DISCUSSIONS

In this study the initial conditions of the agents are set randomly with uniform probability in the interval $[0, 2\pi)$ since the initial orientations of agents are required not to be biased towards any direction. In other words, the probability of an agent i starting with orientation $\theta_{i1}(0) \in [0, 2\pi)$ is equal to the probability of agent i starting with any other orientation $\theta_{i2}(0) \in [0, 2\pi)$. Therefore, the probability distribution function of the

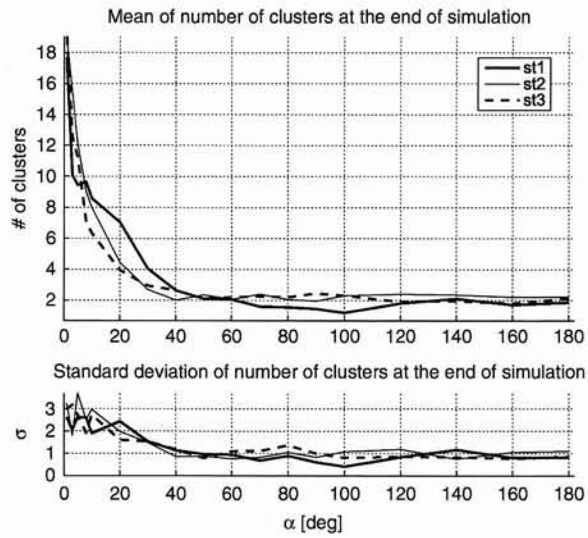


Fig. 24. The number of clusters at $t = T$ for strategies 1 (bold solid line), 2 (solid line), 3 (dash-dot line) for asynchronous case and bounded region.

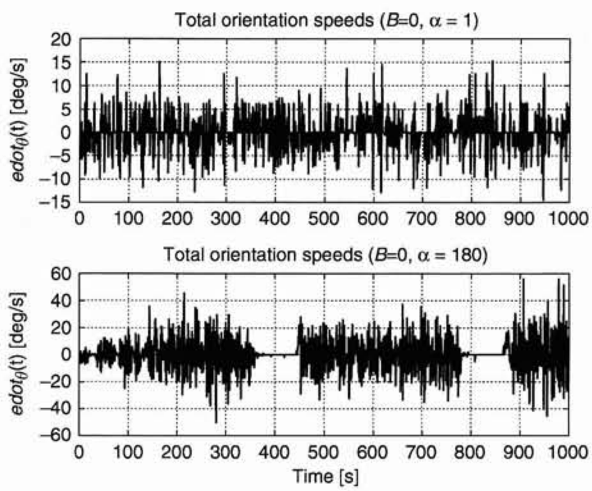


Fig. 25. Total of rate of change of orientations of agents at each step for $\alpha = 1°$ (upper subplot) and $\alpha = 180°$ (lower subplot) for an arbitrary initial condition (Strategy 1 - asynchronous case - bounded region).

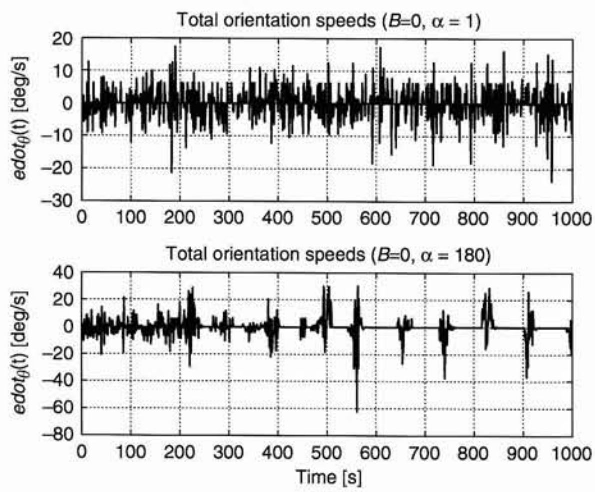


Fig. 26. Total of rate of change of orientations of agents at each step for $\alpha = 1°$ (upper subplot) and $\alpha = 180°$ (lower subplot) for an arbitrary initial condition (Strategy 2 - asynchronous case - bounded region).

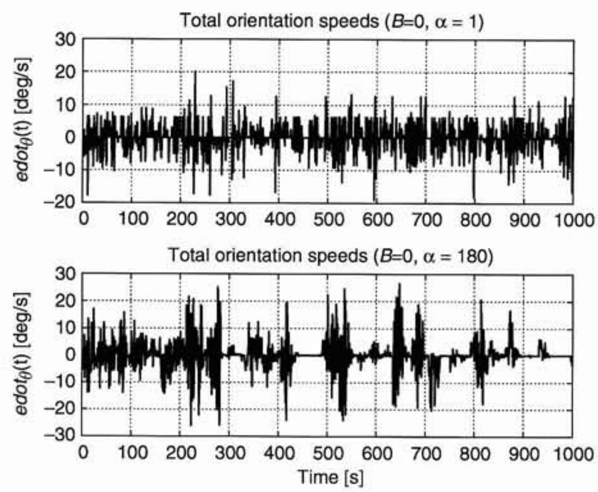


Fig. 27. Total of rate of change of orientations of agents at each step for $\alpha = 1°$ (upper subplot) and $\alpha = 180°$ (lower subplot) for an arbitrary initial condition (Strategy 3 - asynchronous case - bounded region).

initial orientations of the agents in the simulations is given by

$$f(\theta|0, 2\pi) = \begin{cases} \dfrac{1}{2\pi}, & if\ 0 \leq \theta < 2\pi \\ 0, & otherwise. \end{cases} \tag{11}$$

Depending on the initial conditions (orientations and positions), the agents may show different behavior for each of the three strategies. Note that, the group of agents in the neighborhood of agent i, is in fact a sample set of the population distributed uniformly. Considering the asynchronism of updates, the orientations of the set of agents that agent i is utilizing their orientations in performing its orientation update is also a sample set of a uniformly distributed set.

The following analysis is valid for only $t \geq 1$. However, it is sufficient to illustrate the general tendencies of the strategies considered in this article.

In Strategy 1 (the averaging strategy), the agent in any cluster will update its orientation according to

$$\theta_i(t+1) = \frac{1}{N}\sum_{j=1}^{N}\theta_j(t) \tag{12a}$$

where N is the number of the agents in the neighborhood of agent i including itself. (Note that N is different from $|N_i(t)|$ which we used in previous sections. In fact N includes the number of neighbors and agent itself such that $N = |N_i(t)| + 1$.) Recall that the orientations of agents at $t = 1$ in any neighborhood are in fact in a sample set of the uniform distribution we stated above. Therefore, the expected value of the outcome of the averaging rule is the mean of the uniform distribution. In other words, the expected value of the updated orientations for $t = 1$ (initial step) is

$$E[\theta_i(t+1)] = \mu = \frac{2\pi + 0}{2} = \pi. \tag{13}$$

This shows that the agents utilizing the first strategy will tend to orient themselves towards or basically to agree upon an orientation close to π where the orientations of agents are distributed uniformly in the previous state ($t = 1$ in this case). In other words, the agents calculating their new orientation based on the averaging strategy (Strategy 1) will have a bias towards the angle π. This fact will not change even if the topology is not fully connected or synchronous or there are time delays or turn angle restrictions in the system. This is because taking convex combinations between a set of values (and that is what exactly the averaging rule does) cannot lead to values outside the initial set. Therefore, for any subgroup in the swarm the bias will be towards the initial average orientations of the group and as the number of the members in the group increases or the groups join or disjoin this average will tend to be closer to π. In fact, we have noticed in the simulations that the flocks for the first strategy always tend towards the left of the screen (which is the expected result based on the discussions above). The reason for the drawback/shortcoming of flipping directions in this strategy (mentioned before) is exactly due to this tendency towards π. Note also that the averaging strategy uses global orientations. In other words, for its implementation all the agents need to have means to measure global orientations (*e.g.*, each of them needs to have a compass) and they have to agree a priori on a global reference frame (*i.e.*, the compasses have to be calibrated properly). We believe that this is the main reason for the better performance (faster convergence) of the first strategy. The swarm under this strategy seems as a guided swarm with global bias and the model of Strategy 1 might be suitable for such applications. However, for many minimalist multi-agent applications it may not be possible to have global information (*i.e.*, a global reference frame agreed upon a priori). Therefore, for such applications it may not be possible to implement Strategy 1 even though it converges faster.

We would like to also emphasize that if the orientations of agents were drawn from a uniform distribution of angles which is between $(-\pi, \pi]$ (instead of the set $[0, 2\pi)$) then the expected value of the updated orientations would be $\mu = \pi + (-\pi)/2 = 0$. This means that the agents are guided towards 0 of the global reference frame and the qualitative behavior does not change.

In Strategy 2, the agents will update their orientations according to

$$\theta_i(t+1) = \theta_i(t) + \frac{1}{|N_i(t)|} \times \sum_{j=1}^{|N_i(t)|} mod(\theta_j(t) - \theta_i(t) + \pi, 2\pi) - \pi \tag{14}$$

where $|N_i(t)|$ is the number of agents in the neighborhood of agent i (not including agent itself). In (14), the part $\sum_{j=1}^{|N_i(t)|} mod(\theta_j(t) - \theta_i(t) + \pi, 2\pi) - \pi$ is in a sense the estimation of function $u(\theta - \theta_i(t)) = mod(\theta - \theta_i(t) + \pi, 2\pi) - \pi$. For simplicity lets call $\Theta = \theta - \theta_i(t)$. Then $u(\Theta) = mod(\Theta + \pi, 2\pi) - \pi$. The expected value of $u(\Theta)$ is

$$E[u(\Theta)] = \int_{-\infty}^{\infty} u(\Theta)\frac{1}{2\pi}d\Theta \tag{15a}$$

where given the fact that the (absolute) orientation angles are uniformly distributed in the interval $[0, 2\pi)$ (initial–$t = 1$–orientations in this case) one can show that the probability density function of Θ is given by

$$f(\Theta| - \theta_i(t), 2\pi - \theta_i(t)) = \begin{cases} \frac{1}{2\pi}, & if\ -\theta_i(t) \le \theta < 2\pi - \theta_i(t) \\ 0, & otherwise. \end{cases} \tag{16}$$

The function $u(\Theta)$ is a piecewise continuous function

$$u(\Theta) = \begin{cases} \Theta + 2\pi, & if\ -2\pi \le \Theta < -\pi \\ \Theta, & if\ -\pi \le \Theta < \pi \\ \Theta - 2\pi, & if\ \pi \le \Theta < 2\pi \\ 0, & otherwise. \end{cases} \tag{17}$$

Therefore, the expected value of $u(\Theta)$ is given by

$$E[u(\Theta)] = \int_{-2\pi}^{-\pi} (\Theta + 2\pi)\frac{1}{2\pi} d\Theta \tag{18a}$$

$$+ \int_{-\pi}^{\pi} \Theta \frac{1}{2\pi} d\Theta \tag{18b}$$

$$+ \int_{\pi}^{2\pi} (\Theta - 2\pi)\frac{1}{2\pi} d\Theta. \tag{18c}$$

One can simply show that

$$E[u(\Theta)] = 0. \tag{19}$$

Therefore, for the case with fully connected topology and synchronous motion, the expected value of $\theta_i(t+1)$ becomes

$$E[\theta_i(t+1)] = E[\theta_i(t)] + E[u(\Theta)] \tag{20a}$$

$$= \theta_i(t) + E[u(\Theta)] \tag{20b}$$

$$= \theta_i(t) \tag{20c}$$

which implies that agent i utilizing the second strategy will be tending to preserve its previous orientation. Since agent i was chosen arbitrarily, the same will hold for all the agents, which, on the other hand, implies that there is no global reference or bias towards which all the agents tend to converge. This observation explains why the agents perform worse in Strategy 2 compared to Strategy 1. Note that even though Strategy 2 performs worse than Strategy 1, it might be more suitable for many multi-agent applications. This is because first of all it does not require agreement on a global coordinate system between the agents. Second, in real applications in Strategy 1 the agents have to pass their global orientations to each other by means of some kind of inter-agent communication. In contrast, in Strategy 2 the agents can determine themselves the relative orientations (in their local reference frame) of their neighbors by means of local sensing (without a need for inter-agent communication). In fact, even though Strategy 1 has been inspired by natural phenomena such as the global motion of schools of fish, flocks of birds, or swarms of bacteria or synchronization of the flushing of fireflies [25] and has been used to explain such phenomena, it is difficult to imagine that such natural systems operate based on global information (such as Strategy 1) and operation based on local (relative) information (such as Strategy 2) seems more realistic or natural.

For Strategy 3, the next orientation is calculated as

$$\theta_i(t+1) = angle\left(\sum_{j=1}^{N} r_j(t)\right) \tag{21}$$

where N is the number of the agents in the neighborhood of agent i including itself and as one can recall that $r_j(t)$ denotes the unit vector in the direction of $\theta_j(t)$. Therefore, the x and y components of unit vector $r_j(t)$ become

$$x_j(t) = cos(\theta_j(t)) \tag{22a}$$

$$y_j(t) = sin(\theta_j(t)). \tag{22b}$$

Substituting these into equation (21), the next orientation of agents is calculated as

$$\theta_i(t+1) = angle\left(\sum_{j=1}^{N} [x_j(t),\ \ y_j(t)]\right) \tag{23a}$$

$$= angle\left(\left[\sum_{j=1}^{N} (x_j(t)),\ \sum_{j=1}^{N} (y_j(t))\right]\right) \tag{23b}$$

$$= angle\left(\left[\sum_{j=1}^{N} (cos(\theta_j(t))) \sum_{j=1}^{N} (sin(\theta_j(t)))\right]\right) \tag{23c}$$

or

$$\theta_i(t+1) = angle\left(\left[\frac{1}{N}\sum_{j=1}^{N} (cos(\theta_j(t)))\quad \frac{1}{N}\sum_{j=1}^{N} (sin(\theta_j(t)))\right]\right) \tag{24}$$

where $\frac{1}{N}\sum_{j=1}^{N} (cos(\theta_j(t)))$ and $\frac{1}{N}\sum_{j=1}^{N} (sin(\theta_j(t)))$ are the estimations of means of $cos(\theta_j(t))$ and $sin(\theta_j(t))$, respectively. The expected $x = cos(\theta_i(t+1))$ and $y = sin(\theta_i(t+1))$ values of next orientation becomes

$$E[x] = E[cos(\theta_i(t+1))] = E\left[\frac{1}{N}\sum_{j=1}^{N} (cos(\theta_j(t)))\right] \tag{25a}$$

$$E[y] = E[sin(\theta_i(t+1))] = E\left[\frac{1}{N}\sum_{j=1}^{N} (sin(\theta_j(t)))\right]. \tag{25b}$$

To find the estimations of means of the sinusoidal functions of $\theta_j(t)$ we will utilize the following relation

[57, 58]. If θ is uniform in the interval $[0, 2\pi)$, the probability distribution of $\alpha = a\ sin(\theta + \gamma)$ is given by

$$f(\alpha) = \frac{1}{\pi\sqrt{a^2 - \alpha^2}}, \quad -a<y<a. \tag{26}$$

Therefore, the probability distributions of $y = sin(\theta_j(t))$ and $x = cos(\theta_j(t)) = sin(\theta_j(t) + \pi/2)$ are found as

$$f(x) = \frac{1}{\pi\sqrt{1 - x^2}}, \quad -1<x<1 \tag{27a}$$

$$f(y) = \frac{1}{\pi\sqrt{1 - y^2}}, \quad -1<y<1. \tag{27b}$$

The estimation of x is

$$E[x] = \int_{-1}^{1} x \frac{1}{\pi\sqrt{1 - x^2}}\, dx \tag{28a}$$

$$= \frac{1}{2\pi}\sqrt{1 - x^2}\ |_{-1}^{1} \tag{28b}$$

$$= 0. \tag{28c}$$

The same result can be found for the estimation of y as well. Hence, the expected results of means of x and y are 0. This result states that there is no particular expected output for the next orientation since there is no solution for θ satisfying $sin(\theta) = 0$ and $cos(\theta) = 0$. In other words $angle([0, 0]) = \emptyset$. This result shows that there is no guided initial direction or bias for the agents using Strategy 3. In this sense Strategy 3 is similar to Strategy 2. Therefore, as was the case in Strategy 2, it is natural to expect that Strategy 3 as well is worse than Strategy 1 (which has a bias towards π as one can recall). Hence, most of the comments for Strategy 2 hold for Strategy 3 as well.

VII. CONCLUSIONS

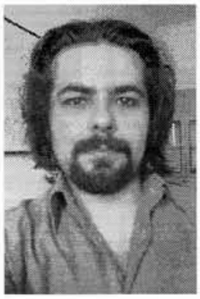
In this study, we compared the performances of three different orientation agreement strategies. We analyzed the behavior of multi-agent systems utilizing these strategies with different combinations of the following properties: (i) the multi-agent systems are synchronous or asynchronous; (ii) they travel in bounded or unbounded regions; and (iii) the mobile agents have various amount of turning speed restrictions. The agents try to orient themselves in the same direction depending on three different interaction rules while at the same time moving with constant speed. We performed each simulation for various turn angle restrictions to observe the effects of non-holonomic dynamics. The agents exhibit best performance in orientation agreement in the case in which they are synchronous, holonomic and using Strategy 1 and in bounded region. In general for all simulations Strategy 1 has the best agreement performance. As discussed in the Section VI we believe that this is mainly due to the fact that in Strategy 1 the agents are guided (*i.e.* have a bias) towards angle π. We showed that if the initial orientations of agents are drawn from a uniform distribution between $[0, 2\pi)$ the orientation updates of agents will be guided towards the angle π of global reference frame for Strategy 1. In contrast, the orientation updates of agents utilizing Strategy 2 and 3 are not guided towards any direction. Each of these strategies have advantages and disadvantages (as discussed in the Discussion section). While Strategy 1 performs best in orientation agreement, for implementation it needs agreement on a global reference frame and measurement of the angles with respect to that frame, whereas Strategy 2 can be implemented using only relative information. For guided orientation agreement problems Strategy 1 would be a good choice. On the other hand, for the cases where global reference frame should not result in bias in orientations of swarms the second and third strategies are better to be utilized.

REFERENCES

1. Cao, Y. U., A. S. Fukunaga, and A. B. Kahng, "Cooperative mobile robotics: Antecedents and directions," *Auton. Robot.*, Vol. 4, No. 1, pp. 7–23 (1997) [Online] Available: overcite.lcs.mit.edu/cao97cooperative.html
2. Mataric, M., "Issues and approaches in the design of collective autonomous agents," *Robot. Auton. Syst.*, Vol. 16, pp. 321–331 (1995) [Online] Available: citeseer.ist.psu.edu/article/mataric95issues.html
3. Bonabeau, E., M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*, Oxford University Press, NY (1999).
4. Kennedy, J. and R. C. Eberhart, *Swarm Intelligence*, Morgan Kaufmann Publisher, San Francisco, CA (2001).
5. Clerc, M. and J. Kennedy, "The particle swarm—explosion, stability, and convergence in a multidimensional complex space," *IEEE Trans. Evol. Comput.*, Vol. 6, No. 1, pp. 58–73 (2002).
6. Passino, K., "Biomimicry of bacterial foraging for distributed optim. and control," *IEEE Contr. Syst. Mag.*, Vol. 22, No. 3, pp. 52–67 (2002).
7. Liu, Y. and K. M. Passino, "Biomimicry of social foraging behavior for distributed optimization: Models, principles, and emergent behaviors,"

J. Optim. Theory Appl., Vol. 115, No. 3, pp. 603–628 (2002).
8. Akyildiz, I., W. Su, Y. Sankarasubramniam, and E. Cayirci, "A survey on sensor networks," *IEEE Commun. Mag.*, No. 8, pp. 102114 (2002).
9. Breder, C. M., "Equations descriptive of fish schools and other animal aggregations," *Ecology*, Vol. 35, No. 3, pp. 361–370 (1954).
10. Warburton, K. and J. Lazarus, "Tendency-distance models of social cohesion in animal groups," *J. Theoretical Biology*, Vol. 150, pp. 473–488 (1991).
11. Okubo, A., "Dynamical aspects of animal grouping: swarms, schools, flocks, and herds," *Adv. Biophys.*, Vol. 22, pp. 1–94 (1986).
12. Grünbaum, D. and A. Okubo, "Modeling social animal aggregations," in *Frontiers in Theoretical Biology*, Lecture Notes in Biomathematics Series, Springer-Verlag, New York, Vol. 100, pp. 296–325 (1994).
13. Grünbaum, D., "Schooling as a strategy for taxis in a noisy environment," *Evol. Ecol.*, Vol. 12, pp. 503–522 (1998).
14. Parrish, J. K., S. V. Viscido, and D. Grünbaum, "Self-organized fish school: An examination of emergent properties," *Biol. Bull.*, Vol. 202, pp. 296–305 (2002).
15. Reynolds, C. W., "Flocks, herds, and schools: A distributed behavioral model," *Comp. Graph.*, Vol. 21, No. 4, pp. 25–34 (1987).
16. Vicsek, T., A. Czirok, E. Ben-Jacob, I. Cohen, and O. Shochet, "Novel type of phase transition in a system of self-driven particles," *Phys. Rev. Lett.*, Vol. 75, No. 6, pp. 1226–1229 (1995).
17. Vicsek, T., A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet, "Novel type of phase transition in a system of self-driven particles," *Phys. Rev. Lett.*, Vol. 75, No. 6, pp. 1226–1229 (1995).
18. Czirók, A., H. E. Stanley, and T. Vicsek, "Spontaneously ordered motion of self-propelled particles," *J. Phys. A Math. Gen.*, Vol. 30, pp. 1375–1385 (1997).
19. Czirok, A. and T. Vicsek, "Collective behavior of interacting self-propelled particles," *Physica A Stat. Mech. Appl.*, Vol. 281, pp. 17–29 (2000).
20. Czirók, A., E. Ben-Jacob, I. Cohen, and T. Vicsek, "Formation of complex bacterial colonies via self-generated vortices," *Phys. Rev. E*, Vol. 54, pp. 1791–1801 (1996).
21. Czirók, A., A.-L. Barabási, and T. Vicsek, "Collective motion of self-propelled particles: Kinetic phase transition in one dimension," *Phys. Rev. Lett.*, Vol. 82, No. 1, pp. 209–212 (1999).
22. Vicsek, T., "Application of statistical mechanics to collective motion in biology," *Physica A Stat. Mech. Appl.*, Vol. 274, pp. 182–189 (1999).
23. Levine, H., W.-J. Rappel, and I. Cohen, "Self-organization in systems of self-propelled particles," *Phys. Rev. E*, Vol. 63, No. 17101, pp. 1–4 (2000).
24. Savkin, A. V., "Coordinated collective motion of groups of autonomous mobile robots: Analysis of vicsek's model," *IEEE Trans. Automat. Contr.*, Vol. 49, No. 6, pp. 981–983 (2004).
25. Jadbabaie, A., J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. Automat. Contr.*, Vol. 48, No. 6, pp. 988–1001 (2003).
26. Moreau, L., "Stability of multiagent systems with time-dependent communication links," *IEEE Trans. Automat. Contr.*, Vol. 50, No. 2, pp. 169–182 (2005).
27. Ren, W. and R. W. Beard, "Consensus seeking in multi-agent systems under dynamically changing interaction topologies," *IEEE Trans. Automat. Contr.*, Vol. 50, No. 5, pp. 655–661 (2005).
28. Tanner, H. G., A. Jadbabaie, and G. J. Pappas, "Stable flocking of mobile agents, part i: Fixed topology," *Proc. Conf. Decis. Contr.*, Maui, Hawaii, December 2003, pp. 2010–2015.
29. Tanner, H. G., A. Jadbabaie, and G. J. Pappas, "Stable flocking of mobile agents, part ii: Dynamic topology," *Proc. Conf. Decis. Contr.*, Maui, Hawaii, December 2003, pp. 2016–2021.
30. Tanner, H. G., A. Jadbabaie, and G. J. Pappas, "Flocking in teams of nonholonomic agents," in Morse, S., N. Leonard, and V. Kumar Eds., *Cooperative Control, Lecture Notes in Control and Information Sciences*, Springer, Vol. 309, pp. 229–239 (2002).
31. Olfati-Saber, R., "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Trans. Automat. Contr.*, Vol. 51, No. 3, pp. 401–420 (2006).
32. Beni, G. and P. Liang, "Pattern reconfiguration in swarms—convergence of a distributed asynchronous and bounded iterative algorithm," *IEEE Trans. Robot. Automat.*, Vol. 12, No. 3, pp. 485–490 (1996).
33. Liu, Y., K. M. Passino, and M. M. Polycarpou, "Stability analysis of one-dimensional asynchronous swarms," *IEEE Trans. Automat. Contr.*, Vol. 48, No. 10, pp. 1848–1854 (2003).
34. Liu, Y., K. M. Passino, and M. M. Polycarpou, "Stability analysis of *m*-dimensional asynchronous swarms with a fixed communication topology," *IEEE Trans. Automat. Contr.*, Vol. 48, No. 1, pp. 76–95 (2003).

35. Gazi, V. and K. M. Passino, "Stability of a one-dimensional discrete-time asynchronous swarm," *IEEE Trans. Syst., Man, Cybern.: Part B*, Vol. 35, No. 4, pp. 834–841 (2005).
36. Beni, G., "Order by disordered action in swarms," *Proc. SAB 2004 Int. Workshop Swarm Robotics*, Lecture Notes in Computer Science (LNCS 3342), Sahin, E. and W. M. Spears, Eds., Springer Verlag, pp. 153–171 (2004).
37. Gordon, N., I. A. Wagner, and A. M. Bruckstein, "Gathering multiple robotic agents with limited sensing capabilities," *Lecture Notes in Computer Science*, Vol. 3172, pp. 142–153 (2004).
38. Flocchini, P., G. Prencipe, N. Santoro, and P. Widmayer, "Gathering of asynchronous oblivious robots with limited visibility," *Lecture Notes in Computer Science*, Vol. 2010, pp. 247–258 (2001).
39. Gazi, V. and K. M. Passino, "Stability analysis of swarms," *IEEE Trans. Automat. Contr.*, Vol. 48, No. 4, pp. 692–697 (2003).
40. Gazi, V. and K. M. Passino, "A class of attraction/repulsion functions for stable swarm aggregations," *Int. J. Contr.*, Vol. 77, No. 18, pp. 1567–1579 (2004).
41. Gazi, V. and K. M. Passino, "Stability analysis of social foraging swarms," *IEEE Trans. Syst., Man, Cybern.: Part B*, Vol. 34, No. 1, pp. 539–557 (2004).
42. Liu, Y. and K. M. Passino, "Stable social foraging swarms in a noisy environment," *IEEE Trans. Automat. Contr.*, Vol. 49, No. 1, pp. 30–44 (2004).
43. Gazi, V., "Swarm aggregations using artificial potentials and sliding mode control," *IEEE Trans. Robot.*, Vol. 21, No. 6, pp. 1208–1214 (2005).
44. Leonard, N. E. and E. Fiorelli, "Virtual leaders, artificial potentials and coordinated control of groups," *Proc. Conf. Decis. Contr.*, Orlando, FL, December 2001, pp. 2968–2973.
45. Bachmayer, R. and N. E. Leonard, "Vehicle networks for gradient descent in a sampled environment," *Proc. Conf. Decis. Contr.*, Las Vegas, Nevada, December 2002, pp. 112–117.
46. Gazi, V., M. I., Köksal, and B. Fidan, "Aggregation in a swarm of non-holonomic agents using artificial potentials and sliding mode control," *Proc. European Contr.*, Kos, Greece, July 2007, pp. 1485–1491.
47. Gazi, V., B. Fidan, Y. S. Hanay, and M. I. Köksal, "Aggregation, foraging, and formation control of swarms with non-holonomic agents using potential functions and sliding mode techniques," *Turkish J. Elect. Eng. Comput. Sci.*, Vol. 15, No. 2, pp. 149–168 (2007).
48. Gazi, V. and B. Fidan, "Coordination and control of multi-agent dynamic systems: Models and approaches," *Proc. SAB06 Workshop Swarm Robotics*, Lecture Notes in Computer Science (LNCS) 4433, Sahin, E., W. M. Spears, and A. F. T. Winfield, Eds., Springer-Verlag, Berlin Heidelberg, pp. 71–102 (2007).
49. Angeli, D. and P.-A. Bliman, "Stability of leaderless multi-agent systems. extension of a result by moreau," arXiv:math.OC/0411338 v1, November 2004.
50. Fang, L., P. J. Antsaklis, and A. Tzimas, "Asynchronous censensus protocols: Preliminary results, simultionsi and open questions," *Proc. Conf. Decis. Contr.* and *European Contr. Conf.*, Sevile, Spain, December 2005, pp. 2194–2199.
51. Blondel, V. D., J. M. Hendrickx, A. Olshevsky, and J. Tsitsiklis, "Convergence in multiagent coordination, consensus, and flocking," *Proc. Conf. Decis. Contr.* and *Proc. European Contro. Conf.*, Sevile, Spain, December 2005, pp. 2996–3000.
52. Cao, M., A. S. Morse, and B. D. O. Anderson, "Agreeing asynchronously: Anouncement of results," *Proc. Conf. Decis. Contr.*, San Diego, CA, USA, December 2006, pp. 4301–4306.
53. Gazi, V., "Stability of a discrete-time asynchronous swarm with time-dependent communication links," *IEEE Trans. Syst., Man, Cybern.: Part B*, Vol. 38, No. 1, pp. 267–274 (2008).
54. Şamiloglu, A. T., V. Gazi, and A. B. Koku, "Effects of asynchronism and neighborhood size on clustering in self-propelled particle systems," *Proc. Int. Symp. Comput. Inform. Sci. (ISCIS06)*, Lecture Notes in Computer Science (LNCS) 4263, in Levi A and et al. Ed., Springer Verlag, Berlin Heidelberg, November 2006, pp. 665–676.
55. Şamiloglu, A. T., V. Gazi, and A. B. Koku, "Asynchronous cyclic pursuit," *Proc. 9th Conf. Simulation Adaptive Behavior (SAB06)*, Lecture Notes in Artificial Intelligence (LNAI) 4095, in S. N. and et al. Ed., Springer Verlag, Berlin Heidelberg, September 2006, pp. 667–678.
56. Bertsekas, D. P. and H. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, Athena Scientific, Belmont, MA (1997).
57. Stark, H. and J. W. Woods, *Probability, Random Processes, and Estimation Theory for Engineers*, Prentice Hall (1994).
58. Papoulis, A., *Probability, Random Variables and Stochastic Processes*, 3rd ed. McGraw-Hill Companies (1991).



Andac T. Samiloglu obtained the B.S. and M.S. degrees in mechanical engineering from the Middle East Technical University, Ankara Turkey in 2003 and 2005, respectively, and is currently working towards his Ph.D. degree at the same department. He is a scholar of the Scientific and Technological Research Council of Turkey (TUBITAK) and the Turkish Academy of Sciences (TUBA). His research interests focus mainly on mechatronics and decentralized coordination and control of multi-agent systems. Since 2006 he has been also an instructor at the Mechanical Engineering Department of Baskent University, Ankara, Turkey.

Veysel Gazi obtained the B.S. degree in electrical and electronics engineering from the Middle East Technical University, Ankara Turkey in 1996 and the M.S. and Ph.D. degrees in electrical engineering from the Ohio State University in 1998 and 2002, respectively. From September 1996 to April 1997 he was a scholar of the Scientific and Technological Research Council of Turkey (TUBITAK). His current research interests focus on decentralized coordination and control of multi-agent dynamical systems. He has coauthored many scientific papers and is the recipient of (shared with Kevin M. Passino) the 2005 Systems, Man, and Cybernetics Society Andy P. Sage Best Transactions Paper Award. He is also co-author (together with M. L. Moore, K. M. Passino, W. P. Shackleford, F. M. Proctor and J. S. Albus) of the book *The RCS Handbook: Tools for Real-Time Control Systems Software Development* published by John Wiley and Sons (Interscience) in 2001. He is serving as a technical publication reviewer for many journals and conferences. He has also served as the Technical Program co-chair of the 2006 Turkish Control Conference and as a guest editor for the ELEKTRIK (Turkish Journal of Electrical Engineering and Computer Sciences) Special Issue on Swarm Robotics published in July 2007.

A. Bugra Koku obtained the B.S. degree in mechanical engineering and M.S. degree in systems and control engineering from the Bogazici University, Istanbul Turkey in 1994 and 1997, respectively. He completed his Ph.D. degree in Electrical Engineering and Computer Science at Vanderbilt University, TN, USA in 2003. His research interests focus mainly on robotics. He is currently conducting research on topics ranging from design of mobile robots to control of outdoor mobile robots, from design of human-like robots to human-robot interaction. Since 2003, he has been teaching at the Mechanical Engineering Department of Middle East Technical University.