This is the Pre-Published Version.

The following publication W. Yuan, N. Ganganath, C.-T. Cheng, S. Valaee, Q. Guo, and F.C.M. Lau, "Semi-flocking-controlled mobile sensor networks for dynamic area coverage and multiple target tracking," IEEE Sensors Journal, vol. 18, no. 21, pp. 8883-8892, 2018 is available at https://dx.doi.org/10.1109/JSEN.2018.2863051.

Semi-Flocking-Controlled Mobile Sensor Networks for Dynamic Area Coverage and Multiple Target Tracking

Wanmai Yuan, *Student Member, IEEE,* Nuwan Ganganath, *Member, IEEE,* Chi-Tsun Cheng, *Member, IEEE,* Guo Qing, *Member, IEEE,* Francis C.M. Lau, *Senior Member, IEEE*

***Abstract*—Mobile sensor networks (MSNs) can provide sensing coverage to large areas of interest (AoIs). Area coverage and target tracking capabilities of MSNs are heavily depending on their motion control and coordination mechanisms. Many existing MSN motion control algorithms ignore or poorly utilize available information from their operating environment, thus lead to unsatisfactory monitoring performances. This paper proposes a fully distributed semi-flocking algorithm which enables mobile nodes to self-organize themselves based on mobility and sensing information via information exchanges among nearby nodes. A distributed mechanism is designed to maximize area coverage and target tracking performances of MSNs. Mobile nodes perform evaluations based on received information and switch between searching and tracking modes. Behaviors of MSNs controlled by the proposed algorithm are studied under different levels of information exchanges. Our study shows that the proposed semi-flocking algorithm is capable of delivering desirable area coverage and target tracking performances in MSNs.**

***Index Terms*—Semi-flocking, mobility sensing, information maps, mobile sensor networks, multi-agent systems.**

I. Introduction

Mobile sensor networks (MSNs), which comprise large numbers of multi-functional mobile nodes, have evolved as a practical option for monitoring and surveillance applications [1]–[5]. With the added mobility over traditional sensor nodes, mobile nodes are capable of relocating and reorganizing themselves to cope with rapidly changing environment and moving targets. Advanced sensing and communication technologies enable nodes to acquire accurate information about their surrounding areas and exchange information among themselves and base stations seamlessly. Local information exchanges allow nodes to have a better understanding of their environment and cooperate effectively.

Common performance criteria of MSNs are their area coverage and target tracking capabilities [6]–[9]. Area coverage is often measured as the portion of the area of interest (AoI) covered under the union of sensing area of mobile nodes at a particular instance or its accumulated value over a finite duration. Full or partial area coverage requirements can be further imposed depending on the nature of the application. In [6], Vecchio *et al.* presented a distributed technique based on a greedy algorithm. In their work, they focused on computing the trajectories of mobile nodes to accomplish a reliable wide-area monitoring system. Apart from area coverage, accurate information on the status of targets, such as their current locations, moving directions, and speeds, is vital to the tracking performance. In [10], Mahboubi *et al.* proposed an energy-efficient strategy for tracking a single moving target with MSNs. Their strategy is based on graph theory and a shortest-path finding algorithm. In their work, their main objective is to reduce the energy consumption of the nodes. To facilitate target tracking, Xu *et al.* [11] and Deshpande *et al.* [12] considered to combine autonomous navigation with target tracking. Others have studied the problem of tracking multiple targets in a given AoI [13], [14].

Many existing works have focused on either area coverage or target tracking separately. However, the multiobjective optimization (MOO) problem involving both area coverage and target tracking lacks sufficient attentions. Potential applications requiring both area coverage and target tracking are numerous, such as search-rescue missions [15], pursuit-evasion games [16], and wildlife-monitoring applications [17]. Compared with their single objective versions, the MOO problem formulation in MSNs has introduced extra requirements and constraints when it comes to algorithm designs. Mobile nodes are then required to switch their operating modes through a multi-criteria decision-making process. Besides, information exchanges and processing among mobile nodes are crucial to their coordination and cooperation.

Heuristic algorithms and techniques are usually employed to yield near-optimal solutions for MOO problems [18]–[20]. Among many heuristic control algorithms, semi-flocking control has demonstrated its outstanding capabilities in satisfying all aforementioned goals in MSNs simultaneously [20]. Semi-flocking algorithms are inspired by the collective and solitary behaviors in natural animal groups. By using simple rules, nodes can steer themselves to perform coordinated actions and achieve multiple purposes of both area sensing and target tracking. The semi-flocking algorithm proposed in [20] has an outstanding surveillance performance. However, their algorithm requires global information on the status of nodes and employs a centralized mechanism to switch nodes between searching and tracking modes. Its centralized design

This work is supported by the Department of Electronic and Information Engineering, the Hong Kong Polytechnic University (Project RUWM).

W. Yuan and Q. Guo are with the School of Electronics and Information Engineering, the Harbin Institute of Technology, Harbin, P.R.China.

N. Ganganath is with the School of Electrical, Electronic and Computer Engineering, the University of Western Australia, WA, Australia.

C.-T. Cheng is with the Department of Manufacturing, Materials and Mechatronics, RMIT University, Melbourne, Australia.

W. Yuan and F.C.M. Lau are with the Department of Electronic and Information Engineering, the Hong Kong Polytechnic University, Hong Kong.

has limited its applicability in many real-world MSNs.

This paper presents a fully distributed semi-flocking algorithm that can fulfill all aforementioned objectives. The main contributions of this paper are summarized as follows:

1) To facilitate distributed coordinations among nodes, information maps and information lists are utilized to record area coverage and target information, respectively. Different levels of information exchanges are studied and carefully analysed in terms of their influences on performances of the proposed algorithm on MSNs.
2) A mode switching mechanism is proposed for nodes to alter between searching and tracking modes. Nodes constantly perform self-evaluation on their sensing performance with information from their neighbors. Each individual uses its evaluation result to determine its next operating mode based on a state transition model.
3) This paper introduces a navigational feedback term that enables nodes to track moving targets with varying accelerations. The proposed switching algorithm further allows nodes to switch back to searching mode when targets suddenly vanished from the AoI.

Extensive tests were conducted to verify the effectiveness of the proposed algorithm. The results show that the proposed distributed semi-flocking algorithm can achieve good performances in both area coverage and multi-target tracking.

The rest of the paper is organized as follows. Section II introduces the distributed information exchange mechanism among nodes. In Section III, the fully distributed semi-flocking algorithm is elaborated. Simulation results are provided in Section IV, followed by their analyses and discussions. Finally, concluding remarks are given in Section V.

II. Local Information Exchange

In this section, we present the preliminaries of MSNs, followed by the formulations of the information maps and the information lists used in recording area coverage and target states, respectively.

A. Preliminaries of MSNs

Consider a MSN consisting of N nodes moving in a rectangle region with width w and length l. The motion of node i is governed by

$$\begin{cases} \dot{q}_i(t) = p_i(t), \\ \dot{p}_i(t) = u_i(t), \qquad i = 1, 2, \ldots, N, \end{cases} \tag{1}$$

where $q_i(t)$ and $p_i(t)$ are the position and velocity of node i at time t, respectively. In (1), $u_i(t)$ is the control input of node i. For notational convenience, we take $q_i(t) = q_i$, $p_i(t) = p_i$, and $u_i(t) = u_i$ as in [19].

While moving in the AoI, a node is able to interact with other nodes within its communication range. The set of neighbors of node i at time t is denoted as

$$\mathcal{N}_i(t) = \{j : \|q_j - q_i\| < r_\mathrm{c}, j = 1, 2, \ldots, N, j \neq i\}. \tag{2}$$

where q_j is the position of node j and $\|\cdot\|$ is the Euclidean norm in $\mathbb{R}^2$ [19]. Here, the communication range r_c is assumed to be identical for all nodes as in [9], [19], [20].

B. Information Maps and Information Lists

Under the proposed algorithm, each node maintains its own information map that records the latest visited time of each sub-region in an AoI. Similar to [9], the AoI is divided into equal-sized cells. Let x_c denote the center of a cell and m_i be the information map of node i. For each node i whose sensing range is r_s, there are three steps in updating its information map.

- Initially, $m_i(x_\mathrm{c}) = 0$ for each cell since area sensing is not yet started before time $t = 0$.
- As node i keeps moving within the AoI, its information map at time t is updated as
$$m_i(x_\mathrm{c}) = t,$$
if $\|x_\mathrm{c} - q_i\| \leq r_s$ for time $t \geq 0$.
- Given $j \in \mathcal{N}_i(t)$, node i can exchange its information map with neighboring node j. Then, $m_i(x_\mathrm{c})$ is updated as
$$m_i(x_\mathrm{c}) = m_j(x_\mathrm{c}),$$
if $m_j(x_c) > m_i(x_c)$. Otherwise, $m_i(x_\mathrm{c})$ remains unchanged.

The first two steps describe how an information map is updated locally via area sensing as time evolves. The third step enables node i to update its information map with its neighboring nodes via local communications. With information maps, nodes can have better understanding on the last visited time of each sub-region in the AoI. Furthermore, nodes can visit areas which have not been visited recently to improve the area coverage performance of the whole MSN.

Apart from sensing the AoI, MSNs controlled by the proposed algorithm can detect and follow multiple targets in the AoI. Multiple targets tracking is a non-trivial task comparing to single target tracking [10]. In this work, an information list is introduced to keep track of locations and movements of each target. When node i discovers a target in its sensing range or via information exchanges with other nodes, node i records the target's information on its information list. Here, we denote l_i as the information list of node i. Consider a target k at q_k^η at time t is moving with a velocity p_k^η and an acceleration a_k^η. Let $l_i(k)$ be the data entry of target k on l_i which includes target k's identity (*i.e.* k), q_k^η, p_k^η, a_k^η, and a time-stamp indicating when $l_i(k)$ was last updated. The updating procedure of $l_i(k)$ is executed as follows:

- Initially (*i.e.* $t = 0$), $l_i(k) = \emptyset$ for any target k and node i within the AoI.
- When $t > 0$ and $\|q_k^\eta - q_i\| \leq r_s$, $l_i(k)$ is updated as
$$l_i(k) = \{k, q_k^\eta, p_k^\eta, a_k^\eta, t\}.$$
- When $t > 0$ and $\|q_k^\eta - q_i\| > r_s$, $l_i(k)$ can only be updated by communicating with other nodes. Suppose node i can make a connection with node j at time $t \geq t_2 > t_1 > 0$, if $l_i(k) = \emptyset$ and $l_j(k) \neq \emptyset$, then $l_i(k)$ is updated as
$$l_i(k) = l_j(k).$$

Alternatively, if $l_i(k) = \{k, q_k^\eta, p_k^\eta, a_k^\eta, t_1\}$ and $l_j(k) = \{k, q_k^\eta, p_k^\eta, a_k^\eta, t_2\}$, then $l_i(k)$ is also updated as

$$l_i(k) = l_j(k).$$

Otherwise, $l_i(k)$ remains unchanged.

Many existing works in MSNs assume nodes only exchange information with neighbors within their communication ranges [19]–[21]. Here, it is assumed that nodes can communicate with each other using store-and-forward method. Three communication scenarios under different levels of information exchange are considered, including a network with fully connected nodes, a network with only 1-hop communications (for both information maps and information lists), and a network with 2-hop (for information lists) and 1-hop (for information maps) communications.

III. Proposed Distributed Semi-flocking

A. Overview of the Proposed Algorithm

Under the proposed algorithm, a node can switch between searching and tracking modes through a distributed mode switching mechanism. For a node in searching mode, it senses the AoI to maximize area coverage according to its information map. The node may update its information map via sensing the AoI or via single-hop communications with its neighbors. Meanwhile, nodes can receive information lists from its own sensor or via β-hop communications with other nodes where $\beta \in \{1, 2\}$. The node performs evaluations on different targets based on the received information lists, then it can decide on whether to track a target or keep on searching the AoI based on its state transition model. The overview of the proposed algorithm is illustrated in Fig. 1, where $n_{\max}$ is a system tuning parameter on defining the maximum allowed number of nodes for tracking a target and n_{ik} represents the number of nodes that are tracking target k at time t. Note that n_{ik} in this work is obtained via local information exchanges, and thus guarantees the distributed nature of the proposed algorithm.

B. Operating Modes of Mobile Nodes

Nodes controlled by the proposed algorithm can operate in either searching mode or tracking mode. For nodes in searching mode, they are searching for new targets by maintaining a high global sensing coverage. These nodes are steered by anti-flocking rules (decentering, collision avoidance, and selfishness). When nodes determine to track targets, they perform coordinated motions and form small groups around the targets. For nodes in tracking mode, they are navigated by flocking rules (centering, collision avoidance, and velocity matching). Therefore, the proposed semi-flocking algorithm defines the control input u_i of node i as

$$u_i = \begin{cases} f_i^g + f_i^d + f_i^s, & \text{node } i \text{ is in searching mode,} \\ f_i^g + f_i^d + f_i^t, & \text{node } i \text{ is in tracking mode.} \end{cases} \tag{3}$$

Here, f_i^g is a gradient-based term [19] and is defined as

$$f_i^g = \phi_\alpha(\|q_j - q_i\|_\sigma) n_{ij}, \tag{4}$$

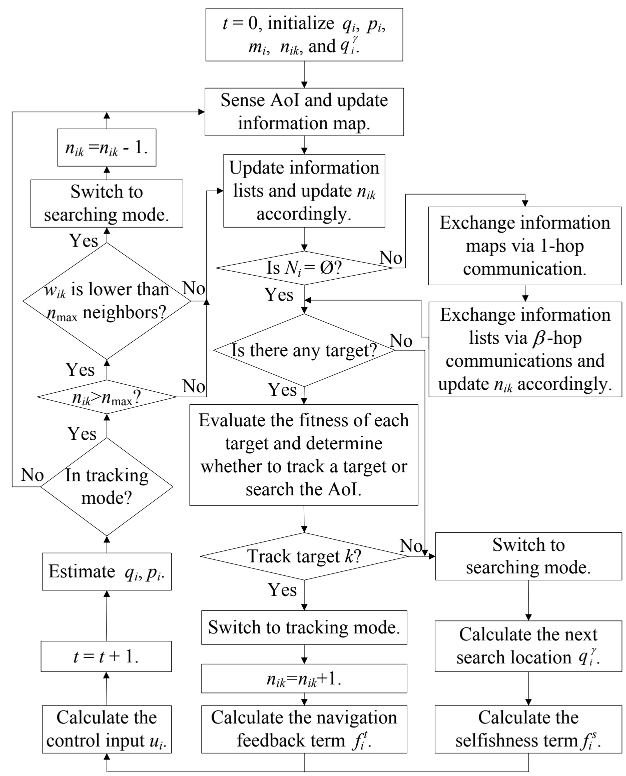


Fig. 1. The overview of the proposed distributed semi-flocking algorithm.

where $n_{ij} = (q_j - q_i)/\sqrt{1 + \epsilon\|q_j - q_i\|^2}$ and $\epsilon \in (0, 1)$. In (4), the action function $\phi_\alpha(z)$ [19] is expressed as

$$\phi_\alpha(z) = \phi(z - d_\alpha)\rho_h\left(\frac{z}{r_\alpha}\right), \tag{5}$$

where

$$\phi(z) = \frac{1}{2}\left[(a + b)\sigma_1(z + c) + (a - b)\right]. \tag{6}$$

In (5), d_α is a constant of α-lattice and $r_\alpha = \|r_c\|_\sigma$, where σ-norm (*i.e.* $\|z\|_\sigma$) of a vector is defined as $\|z\|_\sigma = \left[\sqrt{1 + \epsilon\|z\|^2} - 1\right]/\epsilon$. In (6), $\sigma_1(z) = z/\sqrt{1 + z^2}$. The parameters a, b, and c satisfy $0 < a \le b$ and $c = \mid a - b \mid /\sqrt{4ab}$ [20]. The bump function $\rho_h(z)$ [19] is given as

$$\rho_h(z) = \begin{cases} 1, & \text{if } z \in [0, h), \\ \frac{1}{2}\left[1 + \cos\left(\frac{\pi(z-h)}{1-h}\right)\right], & \text{if } z \in [h, 1], \\ 0, & \text{otherwise,} \end{cases} \tag{7}$$

where $h \in (0, 1)$.

The term f_i^d is a velocity consensus term [19] and is expressed as

$$f_i^d = (p_j - p_i)a_{ij}(q_i, q_j),$$

where p_j is the velocity of node j, and the spatial adjacency matrix $a_{ij}(q_i, q_j)$ [19] is given as

$$a_{ij}(q_i, q_j) = \rho_h\left(\frac{\|q_j - q_i\|_\sigma}{r_\alpha}\right) \quad i \neq j.$$

When nodes are searching the area, these nodes are controlled by $u_i = f_i^g + f_i^d + f_i^s$ in which f_i^s is the selfishness

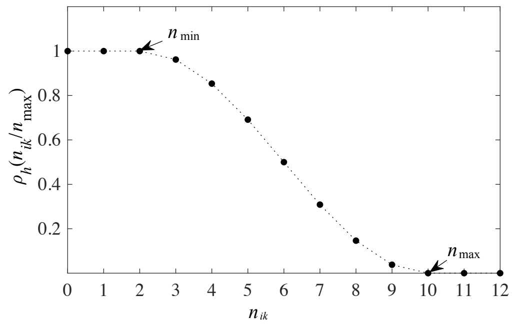


Fig. 2. A bump function $\rho_h(n_{ik}/n_{\max})$ with $n_{\max} = 10$ and $n_{\min} = 2$.

term which is responsible for maximizing the area coverage of each mobile node. In this work, the selfishness term f_i^s is defined as

$$f_i^s = c_1 \frac{(q_i^\gamma - q_i)}{\|q_i^\gamma - q_i\|} - c_2 p_i, \tag{8}$$

where c_1 and c_2 are positive constants. In (8), q_i^γ is the next search location of node i which should be carefully chosen such that the total coverage of the MSN is maximized. The term $\frac{(q_i^\gamma - q_i)}{\|q_i^\gamma - q_i\|}$ is a normalized vector from q_i to q_i^γ. In tracking mode, a node i will follow its target, while its information map will keep being updated. While updating, $q_i^\gamma - q_i$ can become large as q_i^γ can be far away from node i. When the target vanishes from the AoI, if without normalization, a strong force will drive node i to reach q_i^γ which is undesirable. The design rationale is to guarantee nodes to smoothly switch back to searching mode. With the help of distributed information maps (m_i), we adopt the technique proposed in [9] for determining q_i^γ. As mentioned, the AoI is divided into a large number of cells with x_c denoting the center of a cell. Let X_C be the set of all x_c in the AoI. In order to determine q_i^γ, a benefit function $\chi_i(m_i, x_c, t)$ is applied to assess m_i as

$$\chi_i(m_i, x_c, t) = (t - m_i(x_c))(\rho + (1-\rho)\varphi_i(x_c)), \tag{9}$$

where $\rho \in (0, 1)$. The term $(t - m_i(x_c))$ is the time duration after the cell centered at x_c has been last covered by node i. In (9), $\varphi_i(x_c)$ is given by

$$\varphi_i(x_c) = \exp(-\mu_1 \|q_i - x_c\| - \mu_2 \|q_i^\gamma - x_c\|),$$

where μ_1 and μ_2 are positive constants. Here, μ_1 and μ_2 respectively prioritize locations that are close to node i and its current searching location. A node should visit the locations which has the highest benefit value according to its own information map, therefore, q_i^γ is calculated as

$$q_i^\gamma(t+1) = \arg\max_{x_c \in \tilde{X}_C} \chi_i(m_i, x_c, t), \tag{10}$$

where $\tilde{X}_C = \{x_c | x_c \in X_C, \|x_c - q_j\| \geq \|x_c - q_i\| \geq r_s, j \in \mathcal{N}_i(t)\}$.

If node i is currently tracking target k, its navigational feedback term f_i^t is defined as

$$f_i^t = c_3(q_k^\eta - q_i) + c_4(p_k^\eta - p_i) + c_5 a_k^\eta, \tag{11}$$

where c_3, c_4, and c_5 are positive constants. The proposed semi-flocking algorithm with the navigational feedback term introduced in (11) enables nodes to track accelerating or decelerating targets.

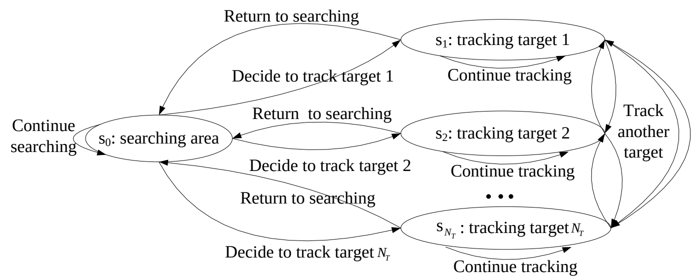


Fig. 3. The state transition model of a node in the proposed semi-flocking algorithm.

C. Mode Switching Mechanism

In this work, a distributed mode switching mechanism is proposed which comprises a self-evaluation method and a multi-state transition model.

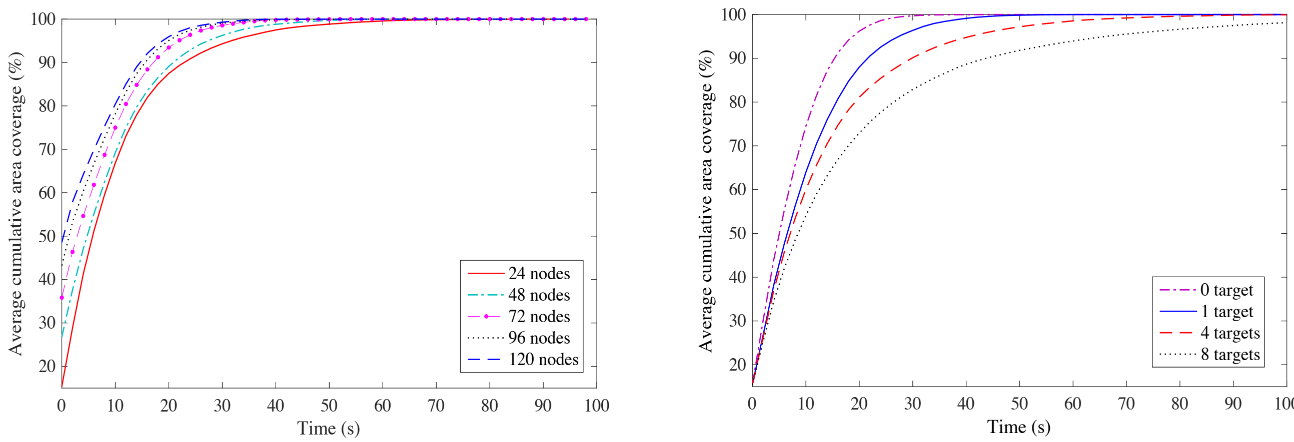
To assess the fitness of a node on different targets, a self-evaluation function for node i at time t is introduced. In this function, apart from data collected by the sensor itself, its number of neighboring nodes in tracking mode, its distances to different targets, and the possible negative impacts due to multi-hop communications have been taken into account. The function is expressed as

$$w_{ik} = \sigma_d \rho_h \left(\frac{n_{ik}}{n_{\max}}\right) \frac{r_s}{\|q_k^\eta - q_i\|}, \tag{12}$$

where w_{ik} denotes the evaluation result of node i for target k. In (12), $\sigma_d \in (0, 1]$ is a generic system parameter which is used to incorporate any possible undesirable impacts due to the multi-hop communications. For illustrative purposes, the value of σ_d is arbitrarily chosen as 1 when a target is within the sensing range of a node, 0.8 when a node receives target information via 1-hop communications, and 0.6 when a node receives target information via 2-hop communications. In practice, values of σ_d should be determined based on the unique properties of the actual system and the targets. Here, $n_{\min}$ denotes the minimum required number of nodes for tracking a target which is subjected to the requirements of applications. As illustrated in Fig. 2, when the number of nodes in tracking mode for target k is lower than $n_{\min}$, the value of $\rho_h(n_{ik}/n_{\max})$ is equal to 1, which indicates target k requires more nodes for tracking it. When $n_{\min} \leq n_{ik} \leq n_{\max}$, the demand on extra tracking nodes will decrease gradually. Finally, when $n_{ik} > n_{\max}$, target k does not need any extra nodes to provide the required coverage.

For the self-evaluation results in (12), the fitness of target k to node i is high when target k has a low number of tracking nodes and is closest to node i. Conversely, low evaluation results are given to distanced targets or targets with sufficient tracking nodes. Such design can make reasonable trade-offs between area sensing and target tracking performances of MSNs. Furthermore, it can regulate the size of the tracking group for each target.

When the information lists from other nodes are received, a node makes its mode switching decision, according to a probabilistic state transition model which is illustrated in Fig. 3. In the model, multiple states are defined to represent cases



(a)

(b)

Fig. 4. Average cumulative area coverage of MSNs with the proposed semi-flocking algorithm (2-hop) (a) with 4 targets and different number of nodes and (b) with 24 nodes and different number of targets. All the data points presented are the results of averaging over 100 simulations.

when a node switches between searching and tracking modes. Initially, all the nodes are initialized in searching mode and continuously searching the AoI. Once a node detects a target and decides to track it, it enters tracking mode, maneuvers toward, and starts following the target. A node returns to searching mode when $n_{ik} > n_{\max}$ or its tracking target disappears from the AoI. The states of a node i are defined as

$$\mathrm{S}_i = \{\mathrm{s}_0, \mathrm{s}_1, \cdots, \mathrm{s}_{N_T}\}, \tag{13}$$

where s_0 denotes the case when the node is in searching mode, while s_1, $\cdots$, s_{N_T} denote cases when the node is tracking targets $\mathrm{T}_1, \cdots, \mathrm{T}_{N_T}$ correspondingly. Note that N_T is the total number of targets in the AoI at time t.

When $\sum_{\zeta=1}^{N_T} w_{i\zeta} \geq 1$, the evaluation result of node i for target k is normalized as

$$w'_{ik} = \frac{w_{ik}}{\sum_{\zeta=1}^{N_T} w_{i\zeta}}. \tag{14}$$

Here, the normalized result w'_{ik} becomes the transition probability for node i in searching mode to switch to track target k as $P_{0k} = w'_{ik}$ $(k \neq 0)$. Meanwhile, $P_{00} = 1 - \sum_{\zeta=1}^{N_T} w'_{i\zeta} = 0$, which means that the node will not remain in searching mode whenever there is a target with a high demand in tracking nodes.

When $\sum_{\zeta=1}^{N_T} w_{i\zeta} < 1$, the transition probability of node i in searching mode to track target k is directly expressed as $P_{0k} = w_{ik}$ $(k \neq 0)$. On the other hand, $P_{00} = 1 - \sum_{\zeta=1}^{N_T} w_{i\zeta} > 0$, which means the transition probability for node i to stay in searching mode is non-zero when the demands for tracking node is moderate.

When $n_{ik} \leq n_{\max}$ and node i meets other targets at time t, the evaluation results of node i on those targets (including its currently tracking target) are normalized as (14). The normalized results w'_{ik} become the transition probabilities P_{ik} for node i in tracking mode to reselect a target k to track. When $n_{ik} > n_{\max}$, the evaluation result of node i for target k is compared with its neighbors' results on the same target via local information exchanges. If there are $n_{\max}$ neighboring nodes whose results are higher than its result, node i return to searching mode or track another target.

TABLE I
PARAMETERS SETTING

Parameters	$n_{\min}$	$n_{\max}$	r_{s}	t_{gap}	ϵ	a	b	h	c_1	c_2	c_3	c_4	c_5	ρ	μ_1	μ_2
Values	3	4	10 m	0.1 s	0.1	5	5	0.2	80	10	2	2	0.15	0.2	0.04	0.01

IV. SIMULATION STUDY

Simulations were conducted to evaluate the performance of the proposed algorithm. To study the effects of information exchanges to the behaviors of the proposed semi-flocking algorithm, systems with fully connected nodes, 1-hop communication, and 2-hop communication were considered, separately. Target tracking and area coverage performances of the proposed semi-flocking algorithm are also compared against the semi-flocking algorithm proposed in [20].

For all the algorithms under test, including the one in [20], initial positions of nodes and targets were selected uniformly at random within a given AoI, while initial velocities of nodes and targets were chosen uniformly at random from the box $[-10, 10]^2$ ms^{-1} and initial accelerations of targets were selected uniformly at random from the box $[-10, 10]^2$ ms^{-2}. The parameters in TABLE I remain constant throughout the simulations for all algorithms under test, where t_{gap} is the time gap of an iteration in the simulations. The rest of the parameters are specified separately with each set of simulations. All the simulations were carried out in MATLAB on a computer with a 2.67 GHz Intel i5 processor, 8 GB memory, and Windows 10 operating system.

A. Area Coverage Performances

The first set of simulations was conducted to analyze the cumulative area coverage of MSNs with the proposed semi-flocking algorithm (2-hop) with different number of nodes ($r_{\mathrm{c}} = 18$ m) and targets. In Fig. 4a, it is observed that MSNs with more nodes can complete a scan faster. As the number

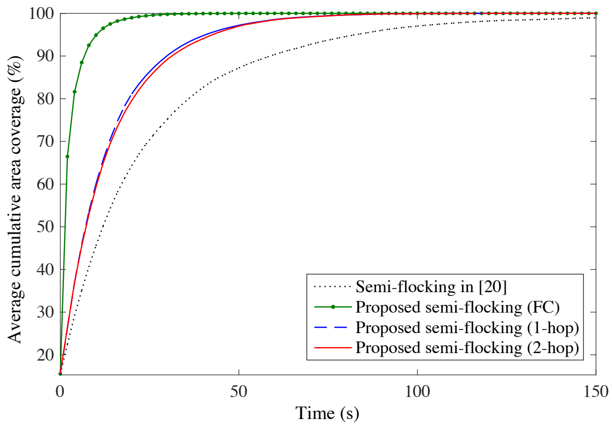


Fig. 5. Average cumulative area coverage of MSNs with four semi-flocking algorithms with 4 targets. All the data points presented are the results of averaging over 100 simulations.

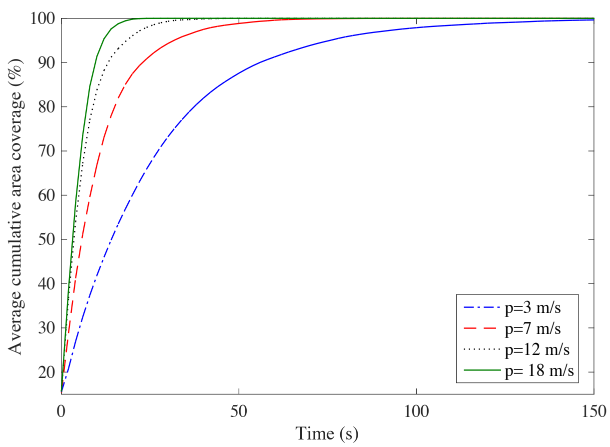


Fig. 6. Average cumulative area coverage of MSNs with the proposed semi-flocking algorithm (2-hop) with 24 nodes and 4 targets. All the data points presented are the results of averaging over 100 simulations.

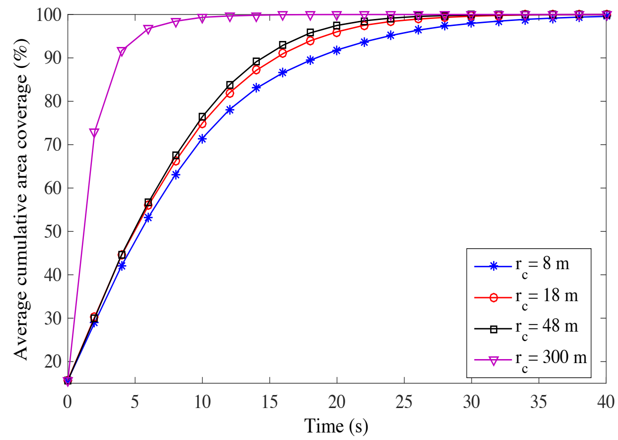


Fig. 7. Average cumulative area coverage of MSNs with the proposed semi-flocking algorithm (2-hop) with 24 nodes and no targets. All the data points presented are the results of averaging over 100 simulations.

of targets is fixed, more nodes operating in searching mode can speed up the search process. Besides, with an increase in the number of nodes, nodes exchange their information more frequently which leads to higher area coverage performances. According to the results in Fig. 4b, cumulative area coverage performances of MSNs degrade with the growth of the number of targets. Which is understandable as there are more targets in the given AoI, more nodes will be operating in tracking mode rather than searching mode.

The next set of simulations was conducted to compare cumulative area coverage performances of MSNs with the proposed algorithm against those with that in [20]. In the simulations, there were 24 nodes with $r_{\mathrm{c}} = 18$ m and 4 randomly moving targets inside the given AoI. According to Fig. 5, as expected, MSNs with fully connected (FC) nodes and the proposed semi-flocking algorithm (*i.e.* the proposed algorithm with $r_c = 300$ m $> \sqrt{2 \times 200^2}$ m in which all nodes in the AoI can communicate with each other) spent the least amount of time to complete a scan of the AoI. By allowing nodes to know better about their environment with the help of the information maps, MSNs with the proposed algorithms can complete a scan faster than those with the one in [20]. According to the results, performance differences among MSNs with the proposed algorithms but with different β values are insignificant. Comparatively, MSNs with the proposed algorithm (1-hop) can complete the scan faster because information lists are only allowed to be exchanged via 1-hop communications under the proposed algorithm (1-hop). As the information is confined to a small set of nodes, a nice balance between nodes in searching mode and tracking mode can be maintained. However, with the proposed algorithm (2-hop), targets' information is being spreaded much farther, which can cause some nodes that are distanced away from the targets to switch into tracking mode.

The third set of simulations was conducted to analyze the cumulative area coverage of MSNs with the proposed semi-flocking algorithm (2-hop) with 24 nodes ($r_{\mathrm{c}} = 18$ m) and 4 targets. In the simulations, the average speeds of nodes (p) were carefully adjusted between 3 ms^{-1} and 18 ms^{-1}. In Fig. 6, as expected, the performance on area coverage degrades with the reduced speed of nodes. When nodes move faster in the AoI, they can explore the AoI faster as they traverse longer distance over the same amount of time. However, if the average speed of nodes are being too high, it may cause collisions in real-world applications. Therefore, such value should be carefully selected based on the unique properties of the mobile nodes, the scenario, and the application.

To analyze the relationship between area coverage and communication range of nodes with the proposed semi-flocking algorithm (2-hop), the fourth set of simulations was conducted with 24 nodes and no targets in the given AoI. Simulation results are given in Fig. 7. As mentioned earlier, MSNs with r_{c} = 300 m can complete scans of the AoI quickly as it enables nodes to form a fully connected network. For r_{c} values ranging from 8 m to 48 m, the cumulative area coverage increases with the rise of r_{c}. Nodes can have better coordinations in scanning the AoI due to the benefits of having larger r_{c}. However, the corresponding improvements to AoI coverage are insignificant. With local communications allowed, information maps are able to be disseminated to the majority of the network

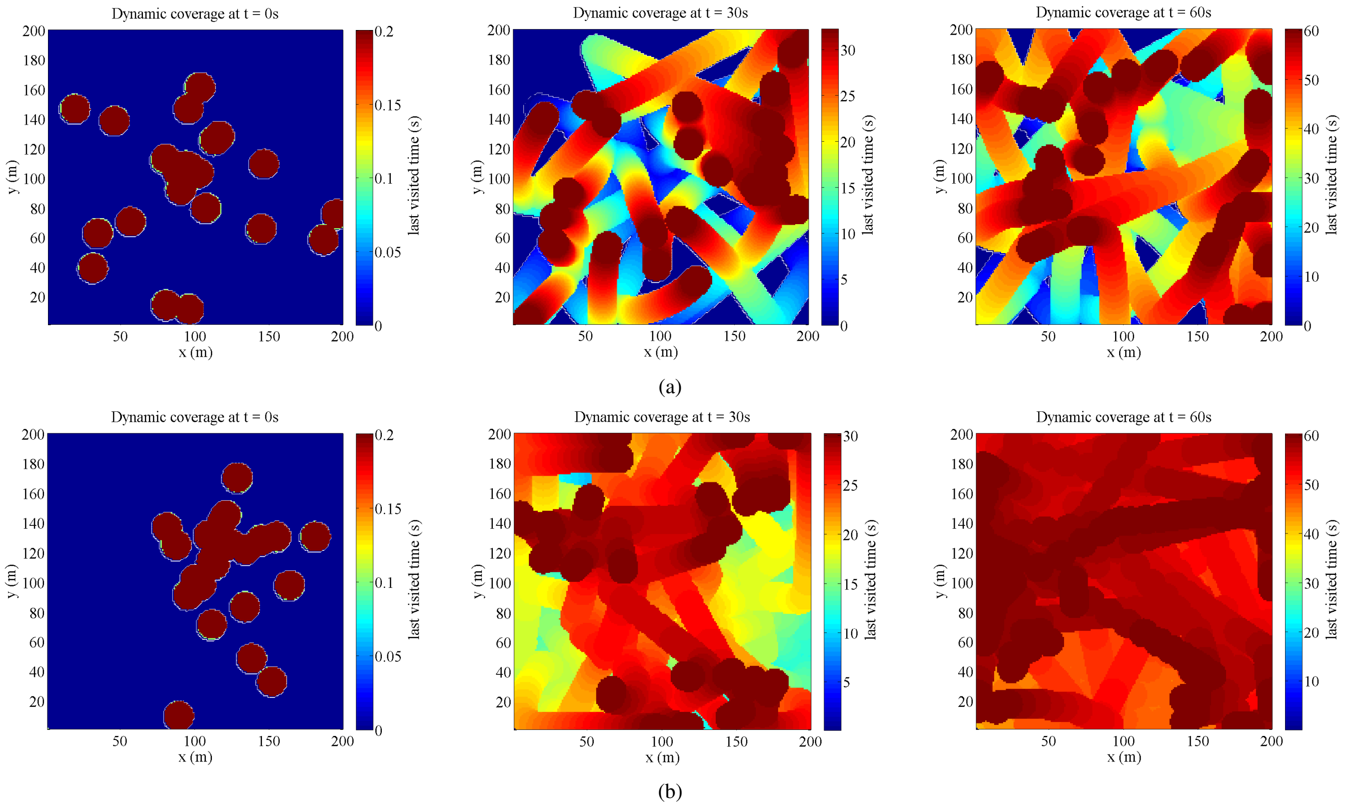


Fig. 8. Dynamic area coverage at time 0, 30, and 60 s of MSNs with (a) the semi-flocking algorithm in [20] and (b) the proposed semi-flocking (2-hop) algorithm.

and allow nodes to search the AoI efficiently even when r_c is relatively short.

Fig. 8 shows the last visited time of the AoI of the 2 algorithms at their 0, 30, and 60 s in two separate simulations. During their initial phases (0-30 s), the proposed algorithm (2-hop) allows nodes to visit almost every cells in the AoI, while for that in [20], a large portion of the AoI is still not visited (indicated in blue). When given a longer duration ($t = 60$ s), MSNs with the proposed algorithm can cover the AoI effectively and almost all the cells in the AoI have been visited within the last 20 s. Comparatively, MSNs with the algorithm in [20] still have some spots being unvisited at $t = 60$ s.

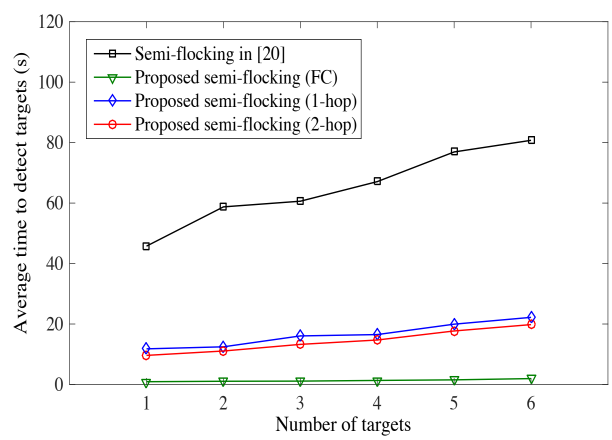


Fig. 9. Average target detection time of MSNs with four semi-flocking algorithms. All the data points presented are the results of averaging over 100 simulations.

B. Target Tracking Performances

The next set of simulations is aimed to demonstrate the target tracking ability of the proposed semi-flocking algorithm. Here, the average detection time of target k is defined as $\sum_{i=1}^{n_{\min}} t_{ik}/n_{\min}$, where t_{ik} is the time when node i decides to track target k. It can be seen from Fig. 9 that the average target detection time increases slightly with the number of targets. It is understandable as the number of nodes is fixed, a longer time is required to detect an increased number of targets. From Fig. 9, it is also observed that the detection time in MSNs with the proposed algorithm is significantly smaller than those with the one in [20]. For the algorithm in [20], due to the limitation of utilizing only the conditions of 8 adjacent areas in making navigational decisions, nodes are often trapped in local optimum points. Most importantly, there is no exchanges on targets' information among nodes, which makes the target detection process close to a random search. On the other hand, by using information maps and lists, MSNs with the proposed algorithm can deliver better performances on both area coverage and target tracking.

Due to the exchange of information lists, nodes located far away from the targets can still be informed through single-hop and/or multi-hop communications, which explains why MSNs with the proposed algorithms (1-hop and 2-hop) can achieve shorter target detection time. According to the results, it is

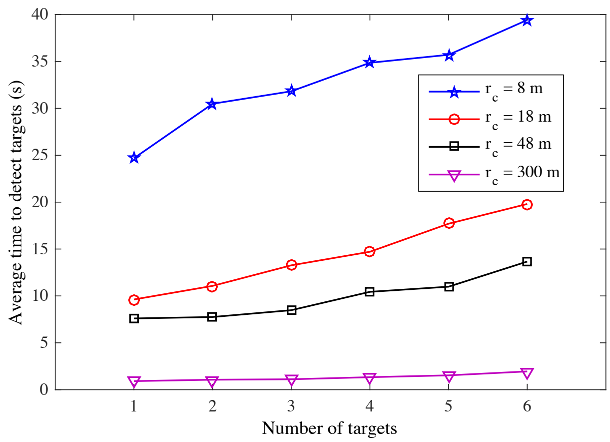


Fig. 10. Average target detection time of MSNs with the proposed semi-flocking algorithms (2-hop) with different communication ranges. All the data points presented are the results of averaging over 100 simulations.

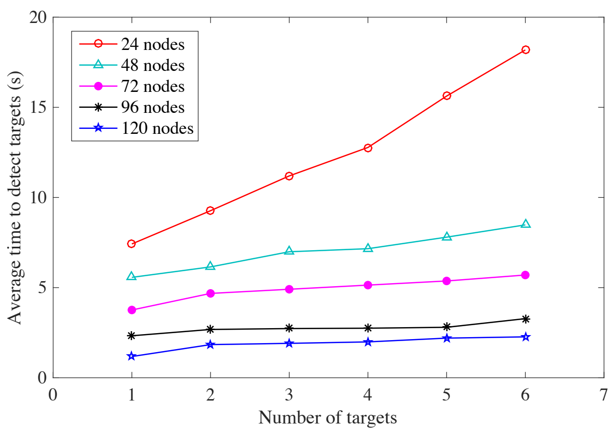


Fig. 11. Average target detection time of MSNs with the proposed semi-flocking algorithms (2-hop) with different number of nodes. All the data points presented are the results of averaging over 100 simulations.

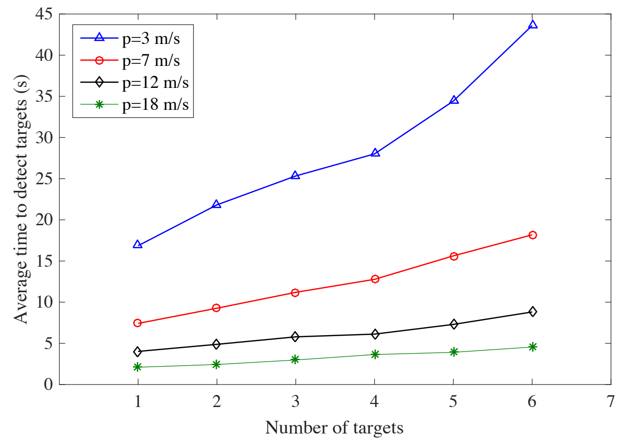


Fig. 12. Average target detection time of MSNs with the proposed semi-flocking algorithms (2-hop) with the different speed of nodes. All the data points presented are the results of averaging over 100 simulations.

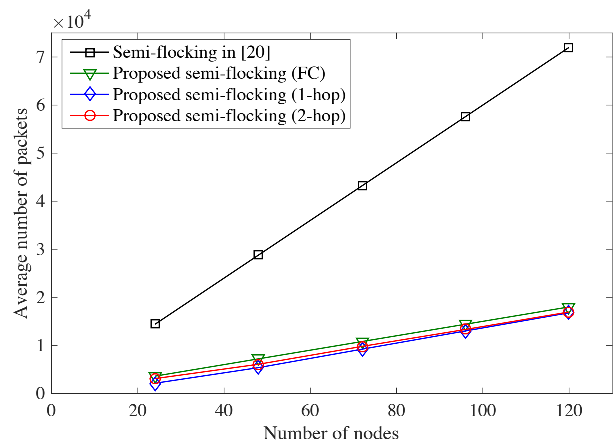


Fig. 13. Average number of packets transmitted in the MSNs with four semi-flocking algorithms with 4 targets and $r_c = 18$ m in time [0,15] s. All the data points presented are the results of averaging over 100 simulations.

observed that the improvement for changing β from 1 to 2 is insignificant. In general, having more information improves target detection capabilities of MSNs. However, having a larger value for β means more nodes that are far away from the targets get informed, but most likely they will decide not to track the targets due to long travel distances. Fig. 10 shows the average target detection time of MSNs with the proposed semi-flocking (2-hop) and r_c values ranging from 8 m to 300 m. With an increases in r_c, more nodes are informed about the targets and thus yield shorter target detection time.

According to Fig. 11, the average target detection time decreases with the growth of the number of nodes in the AoI because the probability of detecting a target increases with the node density. From the results in Fig. 12, it is observed that MSNs with higher average speeds can shorten the average target detection time as well. Nodes that move faster can complete scans and detect targets more rapidly, provided that nodes are having higher acceleration values than their targets.

C. Evaluations of Communication Overheads

This section studies the communication overheads of MSNs with the proposed semi-flocking algorithm (2-hop). Simulations were conducted with 4 targets and the number of nodes varied from 24 to 120 in the given AoI. The proposed semi-flocking algorithm enables nodes to exchange information maps and lists to acquire information on AoI and targets. The communication overheads thus arise due to the exchange of information on area coverage and targets' states. In this set of simulations, nodes disseminate their information via local broadcasting. Each information map or list is transmitted using a separate packet.

According to the simulation results in Fig. 13, given $r_c = 18$ m, the communication overheads of the proposed algorithm increase with the growth of the number of nodes. As node density in the AoI increases, nodes have more opportunities to exchange information with others and thus induce higher communication overheads. Nevertheless, it is clear that the proposed algorithms can help significantly lower communication overheads of MSNs compared to those with the algorithm in [20]. It is because the algorithm in [20] disseminates

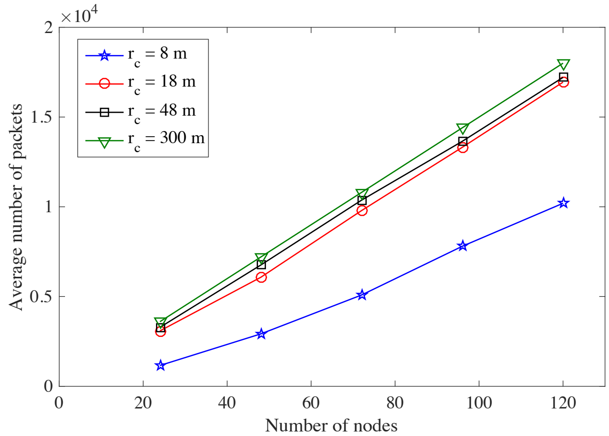


Fig. 14. Average number of packets transmitted in the MSNs with the proposed semi-flocking algorithm (2-hop) with 4 targets in time [0,15] s.. All the data points presented are the results of averaging over 100 simulations.

global information to all its nodes, each individual needs to report its information on area coverage and targets to a central hub and then receive other nodes' information through it. In contrast, nodes controlled by the proposed algorithm exchange information with others via local communications. Most importantly, nodes which do not encounter any target only have to broadcast their information maps, and nodes in tracking mode only have to broadcast their information lists. With such designs, nodes with the proposed algorithm are capable of reducing communication overheads among themselves. Furthermore, it is understandable that MSNs with the proposed algorithm (FC) disseminate more packets compared with those in the proposed algorithm (1-hop and 2-hop). The reason is that nodes controlled by the proposed algorithm (FC) have to broadcast their packets with information maps or lists to all other nodes. In contrast, nodes in the proposed algorithm (2-hop) only need to forward packets with information lists to others via 1-hop and 2-hop communications while their information maps are disseminated only to 1-hop neighbors. From the evaluation results in Fig. 14, as expected, nodes with longer r_{c} have to send and receive more packets with others. It is because when r_{c} is increased, more nodes will fall within the communication range of each other.

V. Conclusion

A fully distributed semi-flocking algorithm for searching and tracking multiple targets with MSNs is proposed in this paper. When compared to an existing algorithm, the proposed semi-flocking algorithm uses distributed information maps and lists, to improve the efficiency and scalability of MSNs in the above applications. Simulation results shown that the proposed semi-flocking algorithm outperforms the existing algorithm in both area coverage and targets tracking capabilities. Most importantly, the proposed algorithm comes with lower communication overheads via utilizing local information exchanges. Future works should focus on estimations of locations, velocities, and accelerations of moving targets using distributed sensor fusion techniques.

References

[1] M. C. Yu and J. S. Leu, "Kernel weighted scheme for improving mobile sensor-node connectivity," *IEEE Sensors Journal*, vol. 13, no. 4, pp. 1200–1206, 2013.

[2] A. R. Al-Ali, I. Zualkernan, and F. Aloul, "A mobile GPRS-sensors array for air pollution monitoring," *IEEE Sensors Journal*, vol. 10, no. 10, pp. 1666–1671, 2010.

[3] M. G. Ball, B. Qela, and S. Wesolkowski, "A review of the use of computational intelligence in the design of military surveillance networks," in *Recent Advances in Computational Intelligence in Defense and Security*. Springer, 2016, pp. 663–693.

[4] J. Yu, S. Wan, X. Cheng, and D. Yu, "Coverage contribution area based k-coverage for wireless sensor networks," *IEEE Transactions on Vehicular Technology*, 2017, Early Access.

[5] H. Kharrufa, H. Al-Kashoash, and A. H. Kemp, "A game theoretic optimization of RPL for mobile Internet of Things applications," *IEEE Sensors Journal*, 2018, Early Access.

[6] M. Vecchio and R. López-Valcarce, "Improving area coverage of wireless sensor networks via controllable mobile nodes: A greedy approach," *Journal of Network and Computer Applications*, vol. 48, pp. 1–13, 2015.

[7] M. Rout and R. Roy, "Self-deployment of randomly scattered mobile sensors to achieve barrier coverage," *IEEE Sensors Journal*, vol. 16, no. 18, pp. 6819–6820, 2016.

[8] J. Du, K. Wang, H. Liu, and D. Guo, "Maximizing the lifetime of k-discrete barrier coverage using mobile sensors," *IEEE Sensors Journal*, vol. 13, no. 12, pp. 4690–4701, 2013.

[9] N. Ganganath, C.-T. Cheng, and C. K. Tse, "Distributed antiflocking algorithms for dynamic coverage of mobile sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 5, pp. 1795–1805, 2016.

[10] H. Mahboubi, W. Masoudimansour, A. G. Aghdam, and K. Sayrafian-Pour, "An energy-efficient target-tracking strategy for mobile sensor networks," *IEEE Transactions on Cybernetics*, vol. 47, no. 2, pp. 511–523, 2017.

[11] E. Xu, Z. Ding, and S. Dasgupta, "Target tracking and mobile sensor navigation in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 1, pp. 177–186, 2013.

[12] N. Deshpande, E. Grant, and T. C. Henderson, "Target localization and autonomous navigation using wireless sensor networks-A Pseudogradient Algorithm Approach," *IEEE Systems Journal*, vol. 8, no. 1, pp. 93–103, 2014.

[13] G. Zhong, S. Niar, A. Prakash, and T. Mitra, "Design of multiple-target tracking system on heterogeneous system-on-chip devices," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 6, pp. 4802–4812, 2016.

[14] R. Ding, M. Yu, H. Oh, and W.-H. Chen, "New multiple-target tracking strategy using domain knowledge and optimization," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 4, pp. 605–616, 2017.

[15] G. Tian, Y. Ren, and M. C. Zhou, "Dual-objective scheduling of rescue vehicles to distinguish forest fires via differential evolution and particle swarm optimization combined algorithm," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 11, pp. 3009–3021, 2016.

[16] C. C. Hsu, Y. Y. Chen, C. F. Chou, and L. Golubchik, "On design of collaborative mobile sensor networks for deadline-sensitive mobile target detection," *IEEE Sensors Journal*, vol. 13, no. 8, pp. 2962–2972, 2013.

[17] L. F. Gonzalez, G. A. Montes, E. Puig, S. Johnson, K. Mengersen, and K. J. Gaston, "Unmanned aerial vehicles (UAVs) and artificial intelligence revolutionizing wildlife monitoring and conservation," *Sensors*, vol. 16, no. 1, p. 97, 2016.

[18] Y.-C. Wang, "A two-phase dispatch heuristic to schedule the movement of multi-attribute mobile sensors in a hybrid wireless sensor network," *IEEE Transactions on Mobile Computing*, vol. 13, no. 4, pp. 709–722, 2014.

[19] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Transactions on Automatic Control*, vol. 51, no. 3, pp. 401–420, 2006.

[20] S. H. Semnani and O. A. Basir, "Semi-flocking algorithm for motion control of mobile sensors in large-scale surveillance systems," *IEEE Transactions on Cybernetics*, vol. 45, no. 1, pp. 129–137, Jan 2015.

[21] N. Ganganath, C.-T. Cheng, and C. K. Tse, "Distributed anti-flocking control for mobile surveillance systems," in *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, May 2015, pp. 1726–1729.