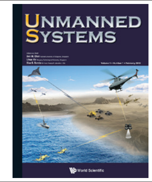
Unmanned Systems, Vol. 5, No. 1 (2017) 31–43

DOI: 10.1142/S2301385017500030

UNMANNED SYSTEMS
http://www.worldscientific.com/worldscinet/us



Multi-Target Engagement in Complex Mobile Surveillance Sensor Networks

Samaneh Hosseini Semnani*, Otman A. Basir†

Electrical and Computer Engineering Department, University of Waterloo, Waterloo, Canada

Efficient use of the network's resources to collect information about objects (events) in a given volume of interest (VOI) is a key challenge in large-scale sensor networks. Multi-sensor multi-target tracking in surveillance applications is an example where the network's success in tracking targets, efficiently and effectively, hinges significantly on the network's ability to allocate the right set of sensors to the right set of targets so as to achieve optimal performance which minimizes the number of uncovered targets. This task can be even more complicated when both the sensors and the targets are mobile. To ensure timely tracking of mobile targets, the surveillance sensor network needs to perform the following tasks in real-time: (i) target-to-sensor allocation; (ii) sensor mobility control and coordination. The computational complexity of these two tasks presents a challenge, particularly in large scale dynamic network applications. This paper proposes a formulation based on the Semi-flocking algorithm and the distributed constraint optimization problem (DCOP). The semi-flocking algorithm performs multi-target motion control and coordination, a DCOP modeling algorithm performs the target engagement task. As will be demonstrated experimentally in the paper, this algorithmic combination provides an effective approach to the multi-sensor/multi-target engagement problem, delivering optimal target coverage as well as maximum sensors utilization.

Keywords: Surveillance systems; biologically inspired computing; constraint optimization problem; semi-flocking algorithm; hierarchical DCOP modeling; sensor coordination; mobility control; target to sensor allocation; sensor networks.

1. Introduction

Sensor networks typically employ a large number of sensory nodes that are capable of collecting massively diverse information about their operating environment. This capability has granted them in recent years an important status in the research community and in a wide range of commercial applications. Their importance is expected to increase as sensor networks continue to advance in capabilities. The use of such networks has proven effective in many applications, including, environmental monitoring, habitat tracking, military command and control, security, surveillance and transportation, to name a few.

An important challenge in large-scale sensor networks is the efficient use of the network's resources to collect information about a given volume of interest (VOI). To address this challenge, sensor networks employ resource management schemes that intended to maximize information gathering at a minimum cost. Often, this goal morphs into a problem of managing a multi-criteria program of conflicting objectives. Achieving such objectives can prove hard, particularly in mission-critical applications such as multi-sensor multi-target surveillance applications where the network's success in tracking targets in a given VOI, efficiently and effectively, hinges significantly on the network's ability to allocate the right set of sensors to the right set of targets so as to achieve optimal performance. This task can be even more complicated if the surveillance application requires the sensors and targets to be mobile.

The surveillance sensor network must maintain engagement with all mobile targets in a given VOI, to ensure timely target tracking. Thus, the network must perform the following real-time tasks alternately at predefined time intervals: (i) combinatorial target-to-sensor allocation; (ii) sensor mobility control and coordination. Given the tasks involved, the sensor-target engagement is obviously

Received 10 May 2016; Revised 19 January 2017; Accepted 20 January 2017; Published 16 March 2017. This paper was recommended for publication in its revised form by editorial board member, Jun Xu.
Email Addresses: **s29hosse@uwaterloo.ca*, †*obasir@uwaterloo.ca*

an NP-hard problem. Thus, for sensor networks to succeed in such an application, an efficient approach that can tackle this NP-hard problem in real-time is needed.

The target-to-sensor allocation problem is concerned with a selection process that involves assigning targets to a set of sensors. This selection process is a combinatorial optimization problem in which some sensors, targets and constraints are given and the goal is to compute an allocation scheme that minimizes constraint violation costs. The more the number of constraint violations, the more the number of uncovered targets. This NP-hard problem requires major computational and communication efforts, especially in large and dynamic sensor networks. Optimization constraints can be used to capture sensor restrictions in tracking a target, for example, to signify the inability of the sensor to track more than one target simultaneously, or the requirement of more than one sensor to track a target.

Generally speaking, surveillance applications imply that targets are mobile, and in this case, the problem definition changes over time. Therefore, even after the sensors find a configuration that enables them to track all targets, the expected dynamism of the environment may quickly invalidate this configuration. Thus, the goal of finding the best assignment of sensors to targets with minimum cost has to adapt to this dynamism by evolving as the environment changes in time. This fact highlights the need for a dynamic sensing strategy whereby sensors optimize mobility control and coordination so as to achieve maximum area and target coverage with minimum cost. The strategy should enable the sensors to continuously plan and coordinate their locations in order to avoid collisions and conflicts.

This paper proposes a new formulation for sensor-to-target engagement in surveillance applications. The formulation captures the issues of sensor-to-target allocation, sensor mobility control, and sensor coordination as intertwined problems. For the sensor-to-target allocation problem, a distributed constraint optimization (DCOP) modeling based approach [1] is employed. Subsequently, the Semi-flocking algorithm [2], a flocking-based approach, is employed for control and coordination.

On one hand, distributed constraint optimization has been widely used for modeling various problems, including sensor-to-target allocation ones [3–8]. This approach has proven to be both general and efficient, especially, in small-scale problems. However, it also tends to be computationally expensive, and often intractable in large-scale problems. The hierarchical DCOP modeling approach proposed in [1] addresses this challenge by decomposing the sensor-to-target allocation problem into smaller sub DCOPs with shared constraints, eliminating significant computation and communication costs.

On the other hand, flocks' self-organizing feature and their ability to benefit from local communication make them suitable for use in the control and coordination of sensors in sensor networks. Flocking-based algorithms such as Flocking, Anti-flocking and Semi-flocking have attracted significant interest in recent years [2], [9–15]. This group of biologically inspired algorithms is characterized by a form of cooperative behavior by a large number of autonomous interacting agents to achieve coordinated group behavior. This behavior is the result of applying a simple set of rules to each sensor while maintaining local communications with neighboring sensors. It has been shown that among flocking-based algorithms, Semi-flocking performs more robust surveillance [2].

The following assumptions have been made in this paper about the surveillance system and mobile sensors:

- The surveillance system consists of n mobile sensors deployed in a two-dimensional geographical region with width w and length l.
- Communication ability: Each sensor can communicate with all its neighboring sensors by exchanging messages through a communication network.
- Sensing ability: Each sensor can sense precise position and velocity of all the targets that are placed within distance r from the sensor. Therefore, the sensing range of each sensor is a circle with radius r around it. Targets that come within this range are always detected, while targets outside are never detected. The problem in which the sensors cannot make accurate measurements is addressed in another paper [25] using distributed Kalman–Consensus filtering. Therefore, in this paper we assume that sensors can make accurate measurements avoiding complication.
- Motion ability: Each sensor's motion is controlled independently but coordinated with the motion of other sensors. Let $q_i, p_i \in \mathbb{R}^2$ denote the position and velocity of sensor i, respectively. The motion of sensor i is governed by the following equation:

$$\begin{cases} \dot{q}_i = p_i \\ \dot{p}_i = u_i, \end{cases} \text{where } q_i, p_i, u_i \in \mathbb{R}^2. \tag{1}$$

  See below for definition of u_i. We assume that sensors are able to move as fast as targets.
- The surveillance system consists of "m" mobile targets ($m < n$), randomly entering and leaving the area of interest (AOI). Each target is assumed to be governed by the following dynamics:

$$\dot{x} = Ax + Bw \quad x \in \mathbb{R}^2, \tag{2}$$

  where x denotes the state of the target, A and B are coefficient time-varing matrixes which define the target dynamics and w is the zero-mean Gaussian noise.
- Knowledge of sensors about targets is limited to targets positions and velocities.

The outline of the paper is as follows. Section 2 provides a brief background on the main DCOP modeling approaches relevant to the target-to-sensor allocation problem. The background focuses on the hierarchical DCOP modeling approach. Section 3 reviews flocking-based algorithms for mobility control of sensors in surveillance applications, with emphasis on the Semi-flocking algorithm. Section 4 describes the proposed sensor engagement formulation. Section 5 presents experimental work conducted to evaluate the proposed formulation. Concluding remarks and recommendations for future work are given in Sec. 6.

2. DCOP Modeling Approaches for Target-to-Sensor Allocation

Distributed constraint satisfaction/optimization (DCSP/DCOP) provides an elegant abstractive formalization of the target-to-sensor allocation problem. In addition, modeling this problem as a DCOP/DCSP facilitates the use of various existing DCSP algorithms as bases for solving the instance at hand. This section introduces DCSP and DCOP briefly and highlights research work reported on the use of this modeling technique to address the target-to-sensor allocation problem, particularly focusing on hierarchical DCOP modeling [1].

2.1. *DCSP/DCOP definition*

DCSP refers to the distributed form of the constraint satisfaction problem (CSP). It is concerned with multiple autonomous agents, each one holding a set of variables. DCSP was first discussed by Sycaro *et al.* [16]. CSP is formally defined as follows [16]:

- A set of n variables: $V = \{X_1, X_2, \ldots, X_n\}$
- A set of finite, discrete domains for each variable: $D = \{D_1, D_2, \ldots, D_n\}$
- A set of constraints: $R = \{R_1, R_2, \ldots, R_m\}$ where each $R_i = (d_{i1}, d_{i2}, \ldots, d_{ij})$ is a predicate defined on the Cartesian product $D_{i1} \times D_{i2}, \cdots \times D_{ij}$. If the value assignment of these variables satisfies this constraint, the predicate returns true; otherwise it returns false.

The final goal of CSP is to find an assignment $A = \{d_1, d_2, \ldots, d_n | d_i \in D_i\}$ that satisfies all the constraints in R.

Although DCSP provides an elegant formulation in many applications, there are still some optimization problems that cannot be modeled as a DCSP. DCOP is a generalized form and optimization version of DCSP, in which the constraints are weighted [17]. In contrast to DCSP, in which each constraint has two states: satisfied or unsatisfied, in DCOP, each constraint has a cost that depends on the values selected for the variables involved, and the goal is to minimize the sum of the constraint costs.

2.2. *Target-to-sensor allocation in DCSP/DCOP modeling*

One important study in solving the target allocation problem in sensor networks has adopted the distributed constraint satisfaction/optimization problem (DCSP/DCOP). Bejar *et al.* [18] introduce two DCSP modeling techniques to model the tracking of mobile targets in wireless sensor networks with static sensors: “sensor centered” and “target centered”. Authors show that these two modeling approaches are duals of each other, and then demonstrate how they differ in their inter-agent and intra-agent communication costs.

Matsui *et al.* in [19,20] and Ota *et al.* in [21] propose two DCOP-based formalizations for the target allocation problem. They introduced two-layer distributed cooperative observation system based on an agency model and then applied the model to a sensor network. First, the agency model was compared with a DCOP-based modeling approach. A cooperative formalization was then proposed to integrate the DCOP approach with the agency one so as to create a combined model. These papers present a more-realistic view of a sensor network that includes autonomous camera sensor nodes, which use pan-tilt-zoom controlled cameras, computers and local area networks and some randomly moving targets. Although they tried for more realistic assumptions, as in a sensor network, targets are beyond the control of the system. The proposed approach does not work because it assigns either targets or (sensor, target)s to variables, and in DCOP, the algorithm can control only the variables. In this approach, the sensors are also considered to be static. In addition, the papers suffer from poor experimental results in which none of the DCOP solvers are applied to solve the problem.

Most of the DCSP/DCOP modeling techniques in sensor network applications suffer from critical and impractical simplifications. These limitations include small network assumptions (e.g., a small number of sensors), static sensor assumptions whereby sensors cannot pursue mobile targets, impractical constraints on the number of targets in the FOV of each sensor and, in some cases, restrictions on target mobility (i.e., static-target assumption). Furthermore, scalability as an important solution aspect has not been considered. As DCSP/DCOP problems are NP-hard [22], raising the number of variables increases the complexity of the problem exponentially. Therefore, using these methods in large-scale sensor networks that consist of a large number of sensors introduces high computational and communicational costs to the DCSP/DCOP algorithm.

Reference [1] presents a technique to overcome these limitations and to eliminate impractical simplifications in such methods.

In this paper we applied the method proposed in [1] for modeling the target-to-sensor allocation problem. This method addresses the problem in two steps:

1. Partitioning the complex target-allocation problem into hierarchically smaller problems.
2. Representing sensors/targets as non-binary variables yielding smaller and more tractable DCOPs.

The partitioning step is based on the geographical locations of the sensors and the environment they monitor. The partitions are constructed such that they have the maximum intra-region constraints, while providing minimum inter-region constraints to make a lighter DCOP in the higher level and as a result to minimize the levels of the hierarchy. Figure 1 depicts the idea schematically.

In the lowest level (Level 1) of the proposed hierarchical architecture, the main problem is partitioned into various regions: Region 0, Region 1, . . . , that everyone is tackled as a distinct DCOP. There are several DCOPs on the lowest level. The next layer (Level 2) of the hierarchy can be tackled as another DCOP that encompasses variables and constraints shared among the connected regions. DCOPs at this multi-region layer do not maintain a view of each region's internal constraints. In fact, only the constraints that exist between different regions are maintained in this layer. Region specific constraints are managed at a lower layer. This partitioning strategy is recursively applied to the sub-partitions, creating a hierarchy of DCOPs of manageable sizes.

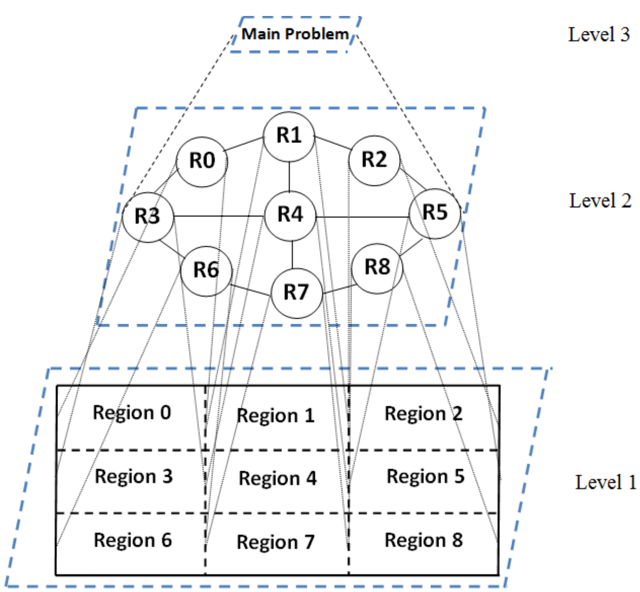


Fig. 1. Hierarchical target-to-sensor allocation model for large and complex sensor networks.

Problem solving using this architecture is a top-down approach. First, the problem of the upper layer is modeled as a DCOP, which is subsequently solved using a DCOP algorithm. Then the lower level's DCOP solvers start solving the problem for every region separately. There are some common variables between the upper and a lower level DCOPs. In solving the problems of the lower level, the algorithm is not concerned with the values maintained by the upper level. Reference [1] mathematically represents that this partitioning decreases the computation cost considerably.

In the second step of this method, non-binary DCOP modeling is applied to model each partition [1]. Previous DCOP techniques [18,20,21] usually represent sensors and/or targets as binary variables, thereby increasing the number of variables and constraints in the DCSP formulation, leading to a larger and more-complex DCSP/DCOP. This could be prohibitive, particularly in large sensor/target problems. Non-binary modeling maps each sensor to a variable and then represents all targets within the FOV of that sensor as that variable's domain. It can also be extended to the target-centered case in which targets are represented as non-binary variables and all the sensors that have that specific target within their FOV construct that variable's domain.

The authors in [1] also selected two DCOP algorithms (ADOPT, DBA) and show how the proposed method can be combined with an appropriate DCOP algorithm. They then used the combination to solve a practical target-to-sensor allocation problem in an airport surveillance system as an example of a large-scale, complex and dynamic sensor network. Their results show that the hierarchical non-binary modeling technique provides better results than the simple non-hierarchical approach.

3. Flocking-Based Algorithms for Mobility Control of Sensors in Surveillance Applications

Flocking-based algorithms are completely distributed algorithms that require only local communications. In this group of algorithms each member applies only a simple flocking rule that has low computation overhead. In addition, they are highly flexible and scalable. All these advantages make flocking-based algorithms a useful approach for tackling motion control problems in a self-organizing surveillance system. This section briefly discusses Flocking, Anti-flocking and Semi-flocking algorithms recently used for mobility control of sensors in a surveillance system, particularly focusing on Semi-flocking algorithm [2], which has been selected in this research for sensor control and coordination.

Flocking algorithm is one of the most successful algorithms in the field of computer science that imitates the behavior of biological creatures. This algorithm is based on three main rules (Reynold)[9]: flock centering which tries to keep each member close to its nearby flock-mates; collision avoidance which tries to avoid collision between flock-mates; and velocity matching which tries to match the velocity of each member with that of its nearby flock-mates. This algorithm has been applied in various application domains, including control and coordination of sensors [10,11,14,23]. In the flocking framework proposed by Olfati-Saber in [10] each sensor (flock) applies a three-term control input function:

$$u_i = f_i^g + f_i^d + f_i^\gamma. \tag{3}$$

This input function contains: a gradient-based term (f_i^g), a velocity consensus term (f_i^d), and a navigational feedback (f_i^γ). The first two terms are related to the three rules of Reynold, and the third one is the flocks' group objective. Equation 4[10] represents the same input function in more detail:

$$u_i = \overbrace{\textstyle\sum_{j\in N_i} \phi_\alpha\left(\|q_j - q_i\|_\sigma\right) n_{ij}}^{\text{Gradient-based term}} + \underbrace{\textstyle\sum_{j\in N_i} a_{ij}(q)(p_j - p_i)}_{\text{Consensus term}} + \underbrace{f_i^\gamma(q_i, p_i, q_r, p_r)}_{\text{navigational feedback}} \tag{4}$$

Applying this simple rule for each sensor results in a network of sensors in which all the sensors keep a specified distance from their neighbors and the whole group moves around the tracking target. Figure 2 shows the mass of sensors created by applying this Flocking algorithm [10]; this algorithm provides robust target coverage for the first target but leaves the other targets uncovered in a multi-target application.

Although the Flocking algorithm provides robust target coverage (for applications with one target), it has serious limitations in providing robust multi-target coverage and also acceptable dynamic area coverage. The Anti-flocking algorithm is an approach presented in [12] to overcome these limitations. This algorithm is based on three main rules (which are different from the three flocking rules): de-centering, which attempts to move each member apart from its neighbors; collision avoidance which attempts to keep members away from their nearest obstacle within a safe distance; selfishness, which moves members toward a direction that maximizes their own gain if neither of two above condition happens. The gain in this paper is defined based on the last time each area is visited by the sensor [12].

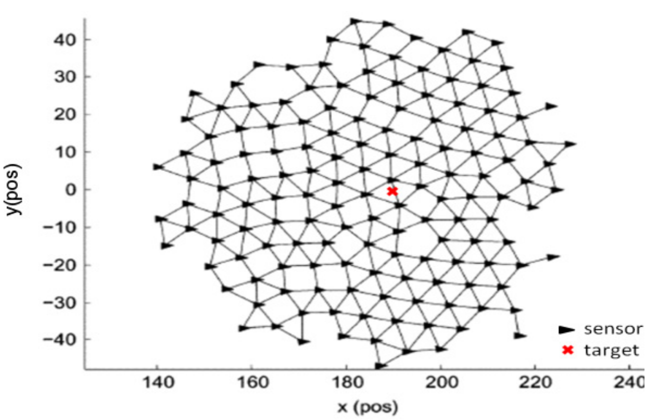


Fig. 2. Flocking for $n = 100$ agent [10].

Anti-flocking algorithm creates an extensive network of sensors, which covers the area of interest dynamically. In this algorithm, the de-centering rule has a great impact in scattering sensors over the whole volume of the monitoring area. Although Anti-flocking algorithm provides acceptable area coverage that helps in timely detection of targets, the lack of adequate sensors flocking around the target in this algorithm greatly increases the network's chance of missing already detected targets.

Semi-flocking algorithm was presented in [2] to overcome the limitations of both Flocking and Anti-flocking algorithms. This algorithm creates a flock of sensors around each target smaller than the big flock that is formed in Flocking algorithm. It also leaves some sensors free to search the volume of the monitoring area for new targets. This behavior in Semi-flocking algorithm is the result of applying an input function u_i by each sensor, which is similar to the control function represented in Eq. (3), except that it makes an essential modification in the third term (f_i^γ) to modify the navigational method of the Flocking algorithm.

The navigational part of the input vector in Semi-flocking algorithm may direct each sensor toward one of the following options:

1. Surrounding targets
2. The least-visited surrounding area

The distance between each sensor and the targets, and the number of sensors tracking each target, are two parameters that have significant impact on directing a sensor toward one of the above options. In this algorithm, if there is one target or more within distance θ to sensor *i*, then it will apply Eq. (5) to navigate toward one of the surrounding targets. Otherwise, the sensor applies Eq. (6) to navigate toward one of the surrounding areas.

$$\begin{aligned} u_i^\gamma &= f_i^\gamma(q_i, p_i, q_{t1}, q_{t2}, \ldots, q_{tm}) \\ &= \sum_{j=1}^{m} \frac{c_{1j}(q_{tj} - q_i)}{n_{tj}} + \sum_{j=1}^{m} \frac{c_{2j}(p_{tj} - p_i)}{n_{tj}} \\ &= \sum_{j=1}^{m} \frac{c_{1j}(q_{tj} - q_i) + c_{2j}(p_{tj} - p_i)}{n_{tj}}, \end{aligned} \tag{5}$$

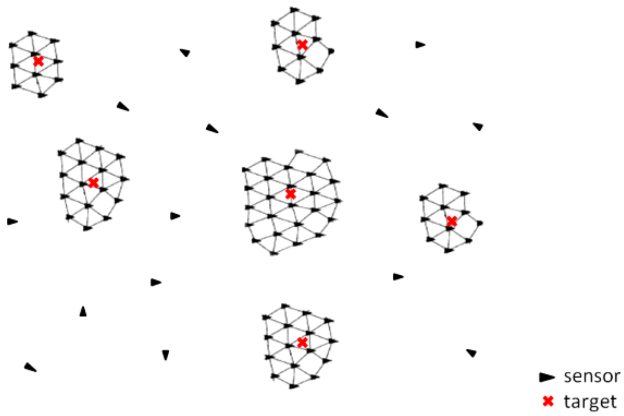


Fig. 3. Semi-Flocking algorithm for sensor management (see Ref. 2).

where $q_{t1}, q_{t2}, \ldots, q_{tm}$ represent the position of targets that are closer than threshold θ to sensor i; c_{1j}, c_{2j} are positive constant values; $q_{tj} - q_i$ is a vector along a line connecting sensor i to target tj; n_{tj} represents the number of sensors currently tracking target tj; and $p_{tj} - p_i$ represents the difference between the velocity of sensor i and target tj.

$$u_i^{\gamma} = c \times (q_{w,l} - q_i), \tag{6}$$

where $q_{w,l}$ represents the center of the least-visited adjacent area, then $q_{w,l} - q_i$ represents a vector along the line connecting the current position of the sensor to the center of the least-visited area; c is a positive constant value that adjusts the size of the vector. Figure 3 represents the position of sensors in Semi-flocking algorithm [2].

The results reported in [2] show that Semi-flocking algorithm has several advantages over other flocking-based methods regarding both target and area coverage.

4. Coupled Allocation and Control Algorithm

Although both hierarchical DCOP modeling and Semi-flocking algorithms are successful approaches, neither is able to address target engagement individually in complex problems. The efficiency of each method strongly depends on the complexity of the problem to be solved. This paper introduces three complexity levels:

- Level 1: Only static sensors (or mobile sensors that are moving randomly and out of any control) are available in the sensor network, and the number of targets is high (static sensors/crowded area).
- Level 2: Mobile sensors (that are moving under a control strategy) are available, but the number of targets is low (mobile sensors/non-crowded area).
- Level 3: Mobile sensors (that are moving under a control strategy) are available and the number of targets is high (mobile sensors/crowded area).

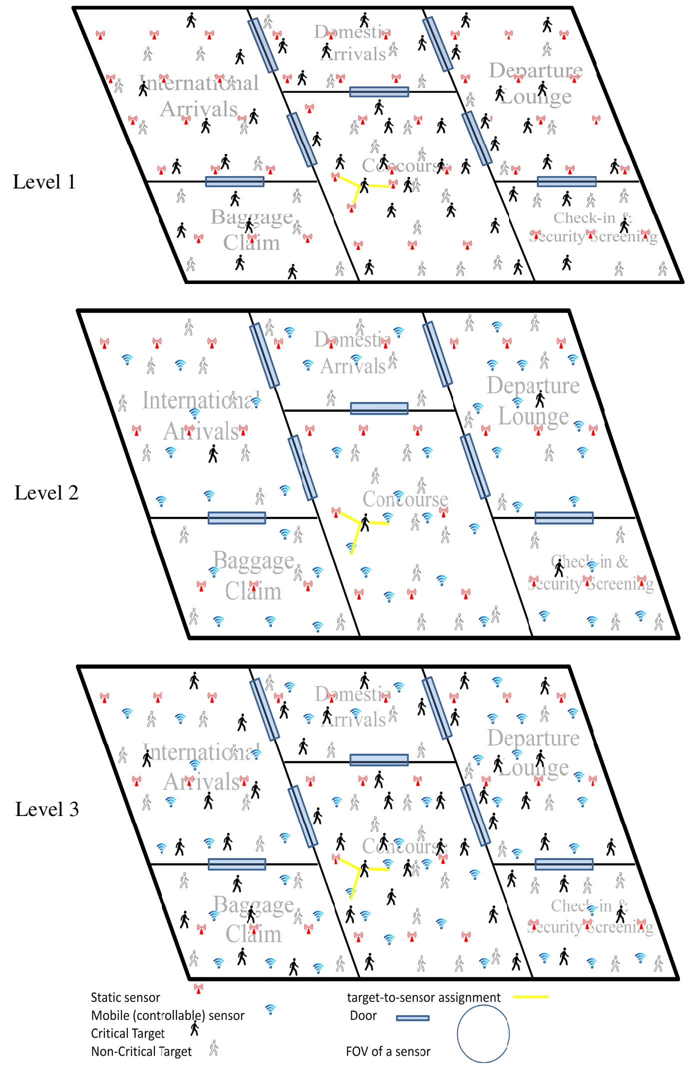


Fig. 4. Target engagement problem in an airport surveillance application. Level 1) non-controllable sensors/crowded area, Level 2) controllable sensors/non-crowded area, Level 3) controllable sensors/crowded area.

Figure 4 depicts these three scenarios in a surveillance system that is used for monitoring a small size airport (i.e., Waterloo Regional Airport). As illustrated in this figure, this surveillance system includes some static and mobile sensors in addition to some critical and non-critical mobile targets.

In all the scenarios the goal is to manage the sensors to maximize (critical) target and area coverage. As represented in this figure, in the first level there are a high number of critical targets, thus it is a crowded and dense area. Also there are a number of static (or mobile but non-controllable) sensors. In this scenario the hierarchical DCOP modeling target-to-sensor allocation approach can be applied to find the optimal assignment between targets and available sensors. Paper [1] represents how this assignment can be done optimally. As the sensors are static or non-controllable, there is no requirement for a control and coordination strategy. In the second level there are both mobile (controllable) and static sensors and also both critical and non-critical targets, but the number of critical targets is very low. Thus, this area can be categorized as a non-crowded area. As there are sufficient sensors, the target engagement problem can be solved using an appropriate control and coordination algorithm, and there is no need for a target-to-sensor allocation approach. As represented in [2], the Semi-flocking algorithm works for such applications.

Finally, the third level describes the most complex scenario, in which there are both mobile (controllable) and static sensors, so there is an essential need for a control and coordination strategy. On the other hand, there are a high number of critical targets, thus there is a high demand on the limited available sensors and we need to apply an appropriate target-to-sensor allocation approach to find the optimal allocation between the targets and sensors. Target engagement in such a complex set-up is a twofold problem that should handle control and coordination of sensors and optimal assignment of sensors to targets, actively. In addition, as targets are mobile, an optimal assignment between sensors and targets may not work over time. Thus, we need to alternately switch between control and allocation strategies. Figure 5 represents target engagement as a twofold problem, where target engagement actively switches between two sub-problems. This figure also illustrates selected approaches to deal with each of the sub-problems.

To have a successful engagement between sensors and targets in the described complex surveillance system, this paper proposes a solution strategy in which the following steps should be repeatedly done:

1. Move the sensors toward their optimal positions based on the Semi-flocking algorithm.
2. Find the optimal assignment between sensors and targets based on the hierarchical DCOP modeling technique.

Figure 6 represents two steps of the proposed approach for solving the target-engagement problem in the airport example (Simplified version of Waterloo Regional Airport).

The three parts of this figure represent the state of the problem before and after applying each of the mentioned steps. Various parts of the airport are represented in Fig. 6(a).

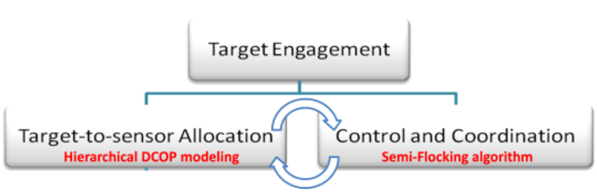


Fig. 5. Target engagement problem as a twofold problem.

This figure illustrates a crowded area including a set of mobile targets and a set of controllable sensors deployed randomly in the surveillance area. Therefore this example is a representative of Level III of the problem depicted in Fig. 4.

In the first step, the Semi-flocking algorithm is used to navigate sensors to create a small flock around each target. It also leaves some of the sensors free to search the surveillance area for new targets. If one or more targets are close to another, the algorithm creates a larger flock to cover all of these targets [Fig. 6(b)]. In the second step, the hierarchical DCOP modeling technique is used to find the optimal assignment of 10 sensors to each target based on their current positions [Fig. 6(c)]. As described in [1], it first partitions the main problem into sub-problems and next models each part as a separate non-binary variable DCOP. Then it uses a DCOP algorithm to find the optimal assignment for covering the maximum number of targets.

After finding a suitable assignment using the allocation technique, we need to switch again to the control strategy because of the dynamicity of the engagement problem, which includes the movement of current targets, entering new targets or leaving current targets. If the targets change their positions, then the Semi-flocking algorithm changes the position of the networks formed of sensors to cover the target in its new position. If a new target arrives, the Semi-flocking algorithm shrinks the other groups providing some sensors for the new target. Finally, if a target leaves the AOI, then the Semi-flocking algorithm scatters sensors allocated to that target between other groups in higher demand, or applies them to search the AOI for new targets. Switching between allocation and control strategies continues at predefined time intervals until the end of the monitoring time.

Figure 7 represents the pseudo code presented in this paper for solving the described complex target-engagement problem. This pseudo code includes two parts. In the first part the Semi-flocking algorithm is applied to compute u_i for each sensor i, and then the algorithm changes the speed and position of each sensor based on the calculated value. The next step of this pseudo code describes the hierarchical DCOP modeling, which models the problem based on the new position of the sensors and targets to a hierarchy of DCOPs and then solves the modeled problem using a DCOP solver such as the distributed breakout algorithm (DBA) [24]. The DBA algorithm is selected in this paper because it uses an iterative improvement strategy that is more suitable for real-time tasks such as the problem of this paper. This fact has been shown in [1]. As represented in this figure, switching between Semi-flocking and DCOP modeling continues until the end of the monitoring time. This figure uses the notations listed below:

- L: a variable that shows the level of the hierarchy.
- X: the size of the largest manageable problem.

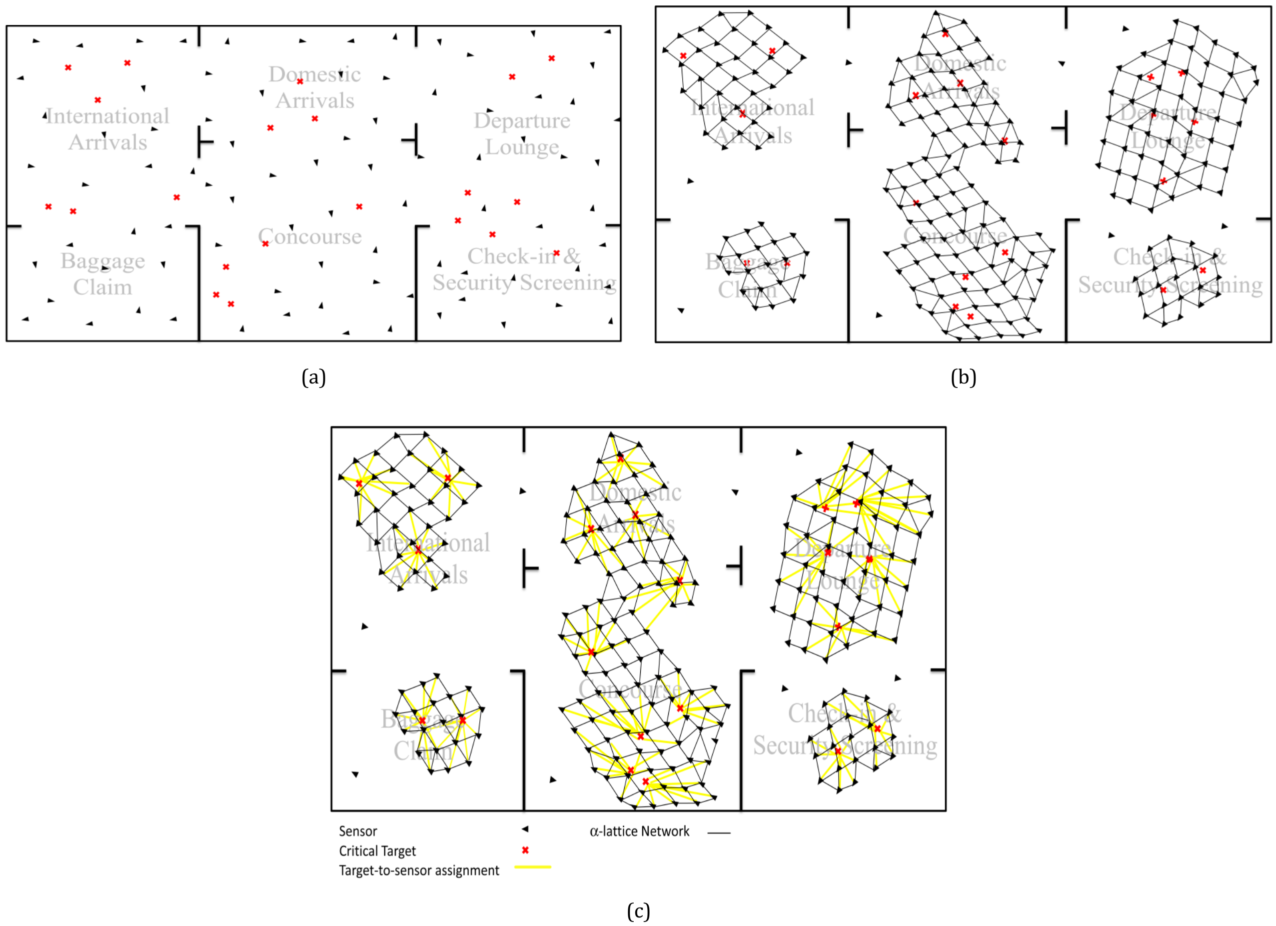


Fig. 6. Example of target engagement using Semi-flocking and hierarchical DCOP modeling approaches, (a) before applying control and allocation algorithms, (b) after applying the Semi-flocking algorithm and (c) after applying the hierarchical DCOP modeling (assignment of 10 sensors to each target).

- P_L: the problem in Level L.
- $P_{L,i}$: the sub-problem i in Level L.
- $Target_{L,i}$: targets in sub-problem i in Level L.
- $Sensor_{L,i}$: sensors in sub-problem i in Level L.
- constraint (P_L): total constraints in level L
- inter-constraint ($P_{L,i}$, $P_{L,j}$): inter region constraints between
- sub-problems $P_{L,i}$ and $P_{L,j}$ in level L.
- intra-constraint ($P_{L,i}$): intra region constraints of $P_{L,i}$ in level L.

5. Experiments and Discussion

5.1. *Experimental setup*

This section describes the experimental setup of the proposed coupled algorithm (Semi-flocking + hierarchical DCOP modeling) for tracking targets in a multi-target environment similar to the small airport described in Sec. 4. In this airport, the area of interest (AOI) is a 1250×665 rectangle, and all the sensors remain in this area for the time $t \geq 0$.

Both parts of the coupled target engagement algorithm (Semi-flocking [2] and hierarchical DCOP modeling (including DBA algorithm) [1] are implemented in Java. These two algorithmic components are executed in predefined time intervals. The parameters of the Semi-flocking algorithm are set as follows: $d = 140, r = 1.2d, \varepsilon = 0.1$ (for σ-norm), $a = 8b, b = 1$ for $\varphi(z), h = 0.2$ for the bump function of $\varphi_\alpha(z)$ The hierarchical DCOP modeling and the DBA algorithm try to find the optimal assignment of three sensors to each target.

In this experiment, the number of targets varied from 15 to 25 and the number of sensors was selected proportional to the number of targets (number of sensors $\cong 6.5 *$ number of targets). The greater the number of targets, the greater the number of sensors deployed in the environment. All the sensors are mobile, and their initial positions are selected randomly.

Control and coordination by semi-flocking algorithm

While (monitoring time is not finished**)**
for sensor $i = 0 \rightarrow n$ **do**
$u_i^{\gamma} = 0;$
for target $t_j = 0 \rightarrow m$ **do**
if $\left\|q_{tj} - q_i\right\|_{\sigma} \leq \theta_j$ **then**
$u_{i,tj}^{\gamma} = \frac{c_{1j}(q_{tj}-q_i)+c_{2j}(p_{tj}-p_i)}{n_{tj}}$
//where n_{tj} is the number of sensors already tracking tj
$u_i^{\gamma} = u_i^{\gamma} + u_{i,t}^{\gamma};$
end if
end for
if $u_i^{\gamma} == 0$ **then** //the sensor is in searching mode
$q_{w,l}$= center of adjacent areas that has least visited times
$u_i^{\gamma} = c \times (q_{w,l} - q_i)$
//toward the area that has longest time not being visited
end if
$u_i = u_i^{\alpha} + u_i^{\gamma}$
$p_i = p_i + u_i \times \Delta t$ //Δt is the updating time intervals
$q_i = q_i + p_i \times \Delta t$ //Δt is the updating time intervals
end for

Target-to-sensor allocation by hierarchical DCOP modeling

Start of modeling
$L \leftarrow 1$
while $|P_L| > X$
partition P_L into a set of sub-problems: $\{P_{L,1}, P_{L,1}, \cdots, P_{L,n}\}$ such that:
a. $\forall i\ i: 1 \rightarrow n$: $|P_{L,i}| \leq X$
b. $\forall i, j\ i, j: 1 \rightarrow n$: min $|inter - constraint(P_{L,i}, P_{L,j})|$
c. $\forall i\ i: 1 \rightarrow n$: max $|intra - constraint(P_{L,i})|$
d. $\forall i, j\ i, j: 1 \rightarrow n$: $P_{L,i} \cap P_{L,j} = \emptyset$
e. $Sensor_L = \bigcup_{\forall i} Sensor_{L,i}$
f. $Target_L = \bigcup_{\forall i} Target_{L,i}$
g. Constraint $(P_L) = (\bigcup_{\forall i} intra - constraint\ P_{L,i}) \cup (\bigcup_{\forall i,j} inter - constraint(P_{L,i}, P_{L,j}))$
for all $P_{L,i}\ i: 1 \rightarrow n$ **do**
model $P_{L,i}$ as a DCOP as described in [1]:
a. $Sensor_{L,i}$ ($/Target_{L,i}$) → variables
b. $Target_{L,i}$ ($/Sensor_{L,i}$) in the FOV of each sensor (/ target) → Domain
c. Intra-constraint($P_{L,i}$) → constraints
end for
$L \leftarrow L + 1$
//all targets that are within FOV of sensors of more than one region participate in the set of targets of next layer
for all $t_i \in Target_{L-1}$ **do**
if $\exists P_{L-1,i}, P_{L-1,j}\ |(FOV(t_i) \cap P_{L-1,i} \neq \emptyset) \wedge (FOV(t_i) \cap P_{L-1,i} \neq \emptyset)$ **then**
add t_i to $Target_L$
end if
end for
//all the sensors that their FOV covers more than one region participate in the set of sensors of next layer

Allocation by hierarchical DCOP modeling

for all $s_i \in Sensor_{L-1}$ **do**
if $\exists P_{L-1,i}, P_{L-1,j}\ |(FOV(s_i) \cap P_{L-1,i} \neq \emptyset) \wedge (FOV(s_i) \cap P_{L-1,i} \neq \emptyset)$ **then**
add s_i to $Sensor_L$
end if
end for
//inter region constraint of previous level constructs constraints of this level
constraint (P_L) = inter-constraint(P_{L-1})
end while
model the last level of hierarchy
model P_L as a DCOP as described in [1]:
a. $Sensor_L$ ($/Target_L$) → variables
b. $Target_L$ ($/Sensor_L$) in the FOV of each sensor (/ target) → Domain
c. constraint(P_L) → constraints
End of modeling
Solve the modeled problem using DBA [24] algorithm
End while

Fig. 7. Pseudocode for solving complex target engagement problem.

The result reported for each point is the average over 10 randomly generated instances. All targets are mobile. In all instances, the initial position and lifetime of each target was selected randomly from a uniform distribution. For each instance, we continued the monitoring time for 300 s (5 min).

The dynamics of targets when they are within the AOI (and are not on the borders) is represented in Eq. (7).

$$x(k+1) = Ax(k) + Bw(k), \tag{7}$$

with $A = \begin{bmatrix} 1 & 0 & \epsilon & 0 \\ 0 & 1 & 0 & \epsilon \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} \epsilon^2/2 & 0 \\ 0 & \epsilon^2/2 \\ \epsilon & 0 \\ 0 & \epsilon \end{bmatrix},$

where $x(k) = (q_1(k), q_2(k), p_1(k), p_2(k))^T$ denotes the state of a target at time k, and $\epsilon = 0.02$ seconds is the step-size. To keep a target within the AOI, the target reverses the previous direction at the borders.

5.2. *Evaluation parameters*

Three parameters were selected to evaluate the proposed coupled algorithm in this experiment:

- *The number of conflicts:* represents the number of constraint violations (conflicts) occurring during the optimal assignment between sensors and targets at each moment of time. A constraint is violated if one of the following conditions happens:

- The number of sensors assigned to a target is less than the required value (in this paper the value of three)
- A sensor is assigned to more than one target

Actually this latter parameter shows how many targets are not perfectly covered by the allocation algorithm.

- *Average of utilization factor:* The utilization factor for each sensor is defined as the ratio of the time that the sensor has been allocated to a target to the total time that has passed since its appearance. Equation (8) represents how this fraction is calculated for sensor *i*. The average of the utilization factor represents the average of the utilization factor over all of the sensors at a specified time.

$$\text{Utilization Factor }(S_i) = \frac{\text{time that } S_i \text{ is allocated to a target}}{\text{current time} - \text{time of the appearance of } S_i \text{ in the FOV}}. \tag{8}$$

- *Variance of utilization factor*: Represents the variance of the utilization factor over all of the sensors at a specified time. The lower the value of variance of the utilization factor means having the more unique pressure on available sensors.

5.3. *Simulation results and analysis*

Figure 8 illustrates a snapshot of the engagement of 98 sensors to 15 targets, using the coupled target engagement algorithm proposed in this paper. In this example, we consider each target to be covered if it is assigned to three sensors at the same time. Yellow lines represent the assignment of a sensor to a target. In this example, the surveillance area is a dense one, and both sensors and targets are mobile. Thus, the coupled algorithm presented is applied to solve the problem. As this figure demonstrates, small flocks are formed and maintained around the targets by the control part (Semi-flocking algorithm). If two or more targets are close to each other, then a bigger flock is created to cover all of them. At the same time, some free sensors explore the AOI. The allocation part (hierarchical DCOP modeling) is assigned three sensors for each target. This assignment is such that the maximum number of targets is covered by the available sensors around them. Note that only sensors within a predefined distance from each target can be assigned to that target. The observations are in close agreement with our expectations of the behavior of this coupled algorithm, depicted in Fig. 6.

The results of evaluating the proposed algorithm are as follows:

(i) *The number of conflicts*: Figure 9 represents the number of constraint violations at each moment of time after running the coupled Semi-flocking and hierarchical DCOP modeling algorithms. This figure demonstrates the results for the problems with 15 to 25 targets. The results show that for all numbers of targets, the optimal assignment found at the beginning of the algorithm contains a high number of conflicts, but as time goes on, the number of conflicts reduces to less than one conflict. This reduction is the result of applying the Semi-flocking algorithm to direct sensors to better positions in which they can be assigned to more targets. In other words, at the beginning, as the sensors are deployed randomly, there are not enough sensors around each target to be assigned to that target by the allocation algorithm; thus, in the first attempts, even a perfect and optimal allocation algorithm cannot find a suitable assignment between sensors and targets due to the lack of sensors in the required positions. Over

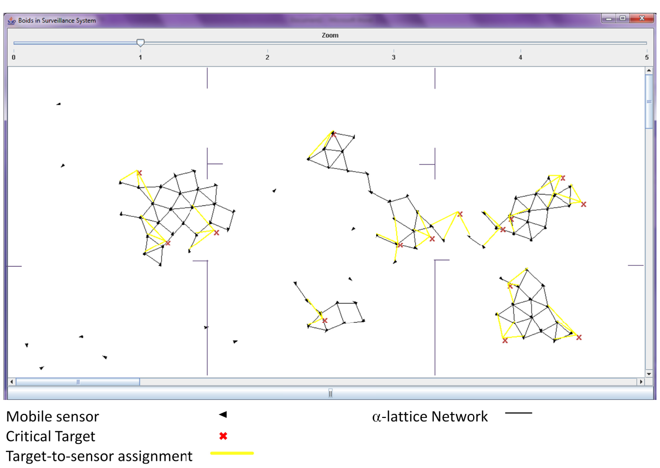


Fig. 8. Snapshot of executing coupled Semi-flocking and hierarchical DCOP modeling approach by 15 targets and 98 sensors.

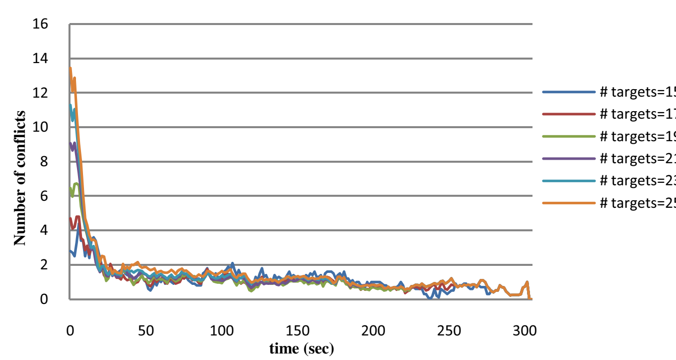


Fig. 9. Number of violated constraints (conflicts) in the coupled Semi-flocking and hierarchical DCOP modeling algorithm.

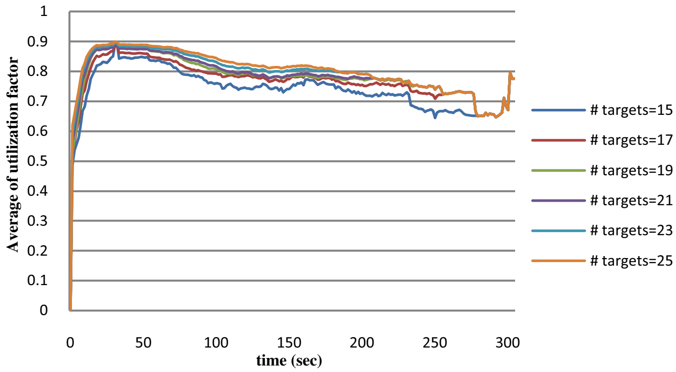


Fig. 10. Average of the utilization factor over all of the sensors in the coupled Semi-flocking and hierarchical DCOP modeling algorithm.

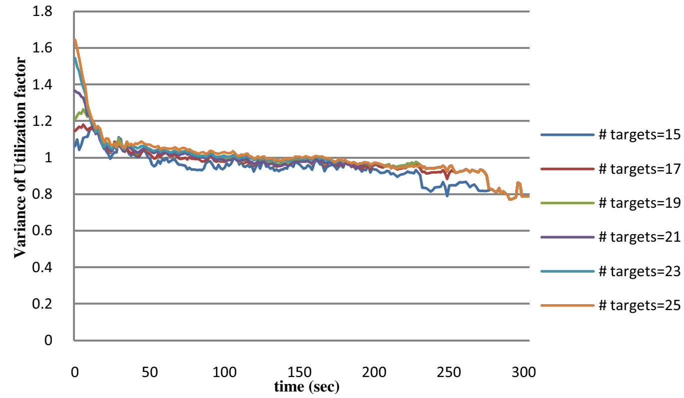


Fig. 11. Variance of the utilization factor over all of the sensors in the coupled Semi-flocking and hierarchical DCOP modeling algorithm.

time, the sensors change their positions based on the Semi-flocking algorithm to the places with higher demand. As a result, a suitable allocation algorithm, such as the one selected in this paper, is able to assign sensors to targets, in a way that avoids most conflicts. The reduction of the conflicts continues to the end of the monitoring time (300 s).

(ii) *Average of utilization factor*: Figure 10 represents the average of the utilization factor over all the sensors at each moment of time for experiments with 15 to 25 targets. As described in part B of this section, the utilization factor for a sensor represents the ratio of the time that the sensors have been allocated to a target, to the total time passed since its appearance. At the beginning of the algorithm, none of the sensors are allocated to any of the targets so their utilization factors are zero. Over time, the number of allocations (due to the changing the sensors positions) and the amount of time that they are allocated, increases and, as a result, the average of utilization factor increases. The highest average for the utilization factor was detected around 50 s after starting the algorithm, but this value does not remain high due to the mobility of the targets, which results in losing some of the assignments and establishing new ones. The utilization factor average remains between the ranges of 0.65 to 0.8 from the 150th second to the end of monitoring time for almost all the experiments (various numbers of targets).

(iii) *Variance of utilization factor*: Figure 11 represents the variance of the utilization factor over all the sensors at each moment of time for experiments with 15 to 25 targets. At the beginning the variance is high as only a few sensors are allocated to a target and the rest are unallocated. As time goes on, a higher number of sensors will be allocated to the targets due to the changing of the sensor positions. As a result, the variance will be decreased. This reduction continues to the end of the monitoring time. A low value of the utilization factor variance in addition to the high value of the utilization factor average represents the fact that numerous targets are allocated to the sensors. It is also important to note that the variance of zero is not an ideal value because we always want to have some sensors free to search the AOI to be available for allocation to the new incoming targets.

(iv) *Scalability analysis*: Although in the experimental analysis we increased the number of targets to 25, it is important to mention that the algorithm does not have any scalability issue when it comes to solve larger problems. The proposed method has two parts. In this section we show how both parts are able to solve larger problems as well as small ones.

As the hierarchical DCOP modeling part uses a partitioning method to divide large scale problems into smaller parts, it is able to successfully avoid large scale DCOP problems. The bigger the problem the more the number of partitions. Using this method, proposed approach is able to solve large size problems as they will be divided into a number of manageble sized problems.

In the control and coordination part, as flocking algorithm is completely distributed algorithm without any centralized part, for solving larger size problems, we just need to increase the number of flocks (sensors). In addition, as in this algorithm, sensors only have local communications with their neighbors, increasing the size of the problem does not have any side effect on increasing the communication complexity.

6. Conclusion

This paper introduced a robust coupled allocation and control algorithm for multiple target engagement, especially in dense surveillance areas in which controllable sensors are available. In this coupled algorithm, Semi-flocking plays the role of controller, and the hierarchical DCOP modeling approach (in addition to the DBA algorithm) acts as the allocator. We show that although each of these algorithms are successful in less-complicated surveillance systems, neither is able to work separately in complex scenarios. The presented pseudo code of the coupled algorithm shows how these two algorithms match with each other and can be applied at predefined time intervals to solve the engagement problem. The proposed algorithm is evaluated over several parameters including the number of constraint violations, which shows the number of non-perfectly covered targets, and the average and variance of the utilization factor. It has been found that the proposed coupled algorithm engages multiple sensors to multiple mobile targets such that the number of uncovered targets decreases efficiency and the sensors' mean utilization factor is increased. These results show that the proposed algorithm has successfully engaged the targets and sensors in dense surveillance systems with controllable sensors and mobile targets.

In the future work, we will consider using the proposed algorithm to find the optimal number of sensors for the specific number of targets to be deployed in the surveillance area. We will look for both theoretical and experimental analysis to find the optimal number and then assess how the results of these two methods match with each other. The optimal number of required sensors is an important parameter for industrial projects and has a great impact in minimizing the costs. Mathematical validation of the proposed method can be considered as another potential future work for this research.

Acknowledgments

This work has been supported by the Ontario Research Fund-Research Excellence (ORE-RE) program through the project "MUltimodal- SurvEillance System for SECurity-RElaTed applications (MUSES SECRET)" funded by the Government of Ontario, Canada.

References

[1] S. Hosseini Semnani and O. Basir, Target to sensor allocation: A hierarchical dynamic distributed constraint optimization approach, *Comput. Commun.* **36** (2013) 1024–1038.

[2] S. Hosseini Semnani and O. Basir Otman, Semi-flocking algorithm for motion control of mobile sensors in large-scale surveillance Systems, *IEEE Trans. Cybern.* (2015) 129–137.

[3] A. Petcu and B. Faltings, A scalable method for multiagent constraint optimization, *International Joint Conf. Artificial Intelligence*, Vol. 19, Lawrence Erlbaum Associates Ltd, (2005), p. 266.

[4] C. Fernandez, R. Béjar, B. Krishnamachari and C. Gomes, Communication and computation in dis. csp algorithms, *Eighth Int. Conf. Principles and Practice of Constraint Programming*, Citeseer, pp. 1–15, 2002.

[5] R. Mailler and V. Lesser, Asynchronous partial overlay: A new algorithm for solving distributed constraint satisfaction problems, *J. Artif. Intell. Res.* **25**(1) (2005) 529–576.

[6] T. Matsui and H. Matsuo, A formalization for distributed cooperative sensor resource allocation, KES-AMSTA, (2008), pp. 292–301.

[7] K. Ota, T. Matsui and H. Matsuo, Layered distributed constraint optimization problem for resource allocation problem in distributed sensor networks, *PRIMA*, (2009), pp. 245–260.

[8] T. Matsui and H. Matsuo, A constraint based formalisation for distributed cooperative sensor resource allocation, *IJIIDS* **4** (2010), pp. 307–321.

[9] C. W. Reynolds, Flocks, herds, and schools: A distributed behavioral model, *Comput. Graph. ACM* Vol. **21**, (1987), pp. 25–34.

[10] R. Olfati-Saber, Flocking for multi-agent dynamic systems: Algorithms and theory, *IEEE Trans. Autom. Control* **51**(3) (2006) 401–420.

[11] R. Olfati-Saber and P. Jalalkamali, Coupled distributed estimation and control for mobile sensor networks, *IEEE Trans. Autom. Control* **57** (2012).

[12] Y. Miao, A. M. Khamis and M. S. Kamel, Applying anti-flocking model in mobile surveillance systems, *AIS* (2010), pp. 1–6.

[13] R. Olfati-Saber, Distributed Kalman filtering for sensor networks, *46th IEEE Conf. Decision and Control* (2007), pp. 5492–5498.

[14] P. Jalalkamali and R. Olfati-Saber, Information-driven self-deployment and dynamic sensor coverage for mobile sensor networks, *American Control Conf. (ACC)* (2012), pp. 4933–4938.

[15] S. Hosseini Semnani, O. Basir and V. Beek, Constrained Clustering for Flocking-based Tracking in Maneuverable Target Environment.

[16] K. Sycara, S. Roth, N. Sadeh and M. Fox, Distributed constrained heuristic search, *IEEE Trans. Syst. Man Cybern.* **21** (1991) 1446–1461.

[17] T. Schiex, H. Fargier and G. Verfaillie, Valued constraint satisfaction problems: Hard and easy problems, *IJCAI* (1995) 631–639.

[18] R. Bejar, B. Krishnamachari, C. Gomes and B. Selman, Distributed constraint satisfaction in a wireless sensor tracking system, Workshop on Distributed Constraint Reasoning, *Int. Joint Conf. Artificial Intelligence*, Seattle, Washington, 2001.

[19] T. Matsui and H. Matsuo, A formalization for distributed cooperative sensor resource allocation, *Agent and Multi-Agent Systems: Technologies and Applications* (2008), pp. 292–301.

[20] T. Matsui and H. Matsuo, A constraint based formalisation for distributed cooperative sensor resource allocation, *Int. J. Intell. Inform. Database Syst.* **4** (2010) 307–321.

[21] K. Ota, T. Matsui and H. Matsuo, Layered distributed constraint optimization problem for resource allocation problem in distributed sensor networks, *Principles of Practice in Multi-Agent Systems*, (2009), pp. 245–260.

[22] P. J. Modi, W.-M. Shen, M. Tambe and M. Yokoo, Adopt: Asynchronous distributed constraint optimization with quality guarantees, *Artif. Intell.* **161**(1–2) (2005) 149–180.

[23] R. Olfati-Saber, Distributed tracking for mobile sensor networks with information-driven mobility, *American Control Conf.,* 2007. *ACC '07*. IEEE, 2007.

[24] K. Hirayama and M. Yokoo, The distributed breakout algorithms, *Artif. Intell.* **161**(1–2) (2005) 89–115.

[25] S. Hosseini Semnani and O. Basir, Robust collaborative multi-target tracking using semi-flocking sensor networks, *IEEE Trans. Autom. Control*.

Samaneh Hosseini Semnani received her B.Sc. degree in Computer Engineering from Kashan University, Kashan, Iran, the M.Sc. degree in Computer Engineering from Isfahan University, Isfahan, Iran, and the Ph.D. degree in Computer Engineering from the University of Waterloo, Waterloo, ON. Her research interests include artificial intelligence, soft computing, sensor networks, and biologically inspired intelligence.

Otman A. Basir (M'09) received his B.Sc. degree in Computer Engineering from Al-Fateh University, Tripoli, Libya, the M.Sc. degree in Electrical Engineering from Queens University, Kingston, ON, Canada, and the Ph.D. degree in Systems Design Engineering from the University of Waterloo, Waterloo, ON. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, University of Waterloo. He is also the Associate Director of the Pattern Analysis and Machine Intelligence Laboratory. His research interests include intelligent transportation systems, embedded real-time systems, sensor networks, sensor and decision fusion, and biologically inspired intelligence.