J Intell Robot Syst (2014) 75:625–640
DOI 10.1007/s10846-013-9887-6

Response Threshold Model Based UAV Search Planning and Task Allocation

Min-Hyuk Kim · Hyeoncheol Baik · Seokcheon Lee

Received: 13 January 2013 / Accepted: 10 September 2013 / Published online: 28 September 2013

Abstract This paper addresses a search planning and task allocation problem for a Unmanned Aerial Vehicle (UAV) team that performs a search and destroy mission in an environment where targets with different values move around. The UAVs are heterogeneous having different sensing and attack capabilities, and carry limited amount of munitions. The objective of the mission is to find targets and eliminate them as quickly as possible considering the values of the targets. In this context, there are two distinct issues that need to be addressed simultaneously: search planning and task allocation. The search plan generates an efficient search path for each UAV to facilitate a fast target detection. The task allocation assigns UAVs attack tasks over detected targets such that each UAV's attack capability is respected. We model these two issues in one framework and propose a distributed approach that utilizes a probabilistic decision making mechanism based on response threshold model. The proposed approach accounts for natural uncertainties in the environment, and provides flexibility, resulting in efficient exploration in the environment and effective allocation of attack tasks. The approach is evaluated in simulation experiments in comparison with other methods, of which results show that our approach outperforms the other methods.

Keywords Unmanned Aerial Vehicles · Search planning · Task allocation · Probabilistic decision making · Response threshold model

M.-H. Kim (✉)
Center for Army Analysis and Simulation, Republic of Korea Army, PO Box 501-15, Bunam-ri, Sindoan-myeon, Gyeryong-si, Chungnam-do, 321-929, South Korea
e-mail: kim507@purdue.edu

H. Baik
School of Aeronautics and Astronautics, Purdue University, 701 W. Stadium Avenue, West Lafayette, IN 47907, USA
e-mail: hbaik@purdue.edu

S. Lee
School of Industrial Engineering, Purdue University, 315 N. Grant Street, West Lafayette, IN 47907, USA
e-mail: lee46@purdue.edu

1 Introduction

The advanced technologies in Unmanned Aircraft System (UAS) allow Unmanned Aerial Vehicles (UAVs) to carry out complex missions that require UAVs to perform various types of tasks [1]. One of crucial missions for which multiple UAVs can be utilized includes a search and destroy mission where a group of UAVs is deployed into a battlefield in order to search for targets and destroy identified targets.

This paper considers simultaneously search planning and task allocation for a group of UAVs that performs a search and destroy mission in a dynamic environment. In the environment, targets move dynamically and are of various types that have different value and munition resources requirement. The value of a target specifies the criticality of the target and may change over time. The munition requirement indicates the minimum amount of munition resources usage required to destroy the target. Each UAV is assumed to have: (1) a target sensing device to detect targets; (2) a communication equipment to exchange information and cooperate with others; and (3) munition resources to attack targets. The UAV team is composed of heterogeneous ones in terms of sensing and attack capabilities. The munition resources that a UAV possesses are limited in quantity and deplete with usage. In a mission, the UAVs are expected to carry out two different types of tasks associated with targets: search and attack. A search task is to search for targets in a search region and an attack task is to strike a target with munition resources. When a UAV detects a target, it either attacks the target single-handedly if it has proper attack capability or assigns it to another UAV if not. The objective of the mission is to find targets and eliminate them as quickly as possible taking into account the values of targets.

In this scenario, there are two important issues that need to be considered simultaneously; search planning and task allocation. The search plan provides each UAV with a search path for fast target detection so that the following attack task over a detected target can be shortly performed. It is inherently impossible to plan an optimal search path for each UAV throughout the mission because of unpredictability in the environment, and hence the search path of each UAV has to be generated in an on-line fashion. The task allocation assigns an UAV to a detected target such that UAV's attack capability and attack response time are respected. Though there have been significant efforts that deal with each issue separately, only few works study the problem that combines these two issues [2].

Motivated by this notion, we model the problem that treats these two distinct issues in one framework. The problem becomes complicated by several factors: multiple UAVs, moving targets, constrained resources, and heterogeneous UAV capabilities. Moreover, tight coupling of search planning and task allocation increases the complexity of the problem. Finding globally optimal solution is computationally impracticable due to the dynamic nature of the problem. Therefore, we need to find an efficient procedure to produce sufficiently good solution to this complex problem. In our work, presented is a distributed approach that utilizes a probabilistic decision making protocol based on the response threshold model, which is inspired by social insects' task assignment behavior [3, 4]. The basic idea is that an individual's behavior is not deterministically but probabilistically determined using a probability function conditioned on two factors, so-called response threshold and stimulus. The response threshold represents the level of preference or specialization of an individual UAV to a task and the stimulus corresponds to the demand, value, or priority of the task. Based on the threshold level and the stimulus intensity, the probability function determines a probability that the individual accepts the task. This probabilistic action selection mechanism accounts for natural uncertainties in the environment and provides flexibility, resulting in efficient exploration in the environment and effective task allocation of attack tasks.

The remainder of this paper is organized as follows. In the next section we present a brief review of related work. In Section 3, we describe our problem in detail. The UAV information base is defined in Section 4, which is followed by approach we propose in Section 5. We show simulation results comparing our approach to other algorithms in Section 6. Finally, Section 7 discusses the conclusion and future work.

2 Related Work

The problem of searching for targets in an environment has been an intensive research area for several decades. The earliest one has been addressed in classical search theory [5], which mainly focuses on the optimal allocation of search resources to maximize detection probability of a stationary target. The basic elements in the optimal search problem include a prior distribution on target location, a function associated with search resource and detection probability, and a constrained amount of search resource. The exponential function is a common assumption to describe the probability of target detection [6, 7]. While the classical search problem concerns a single stationary target [8–10], modern work on search problem has extended the area by considering mobile targets and/or multiple targets. In the work, the search region is partitioned into a finite number of cells in which unknown position of the target is given by a probability distribution. The movement of the target is modeled as a Markovian motion with transition matrix over cells [11]. The transition probabilities of the Markov chain can be constructed based on known dynamics of target, operation strategy of target, and terrain features of the region.

However, all the work mentioned above dealing with a single searcher may not be suitable for modern applications where a team of multiple agents is adopted in order to ensure robustness and faster mission accomplishment. The search problem, when dealing with multiple agents, needs to address several realistic considerations such as heterogeneity of agents and distributed decision making. In recent years, there have been numerous studies on search strategies for multiple agents. Polycarpou et al. [12] address a cognitive map-based cooperative search strategy for a team of UAVs. In their work, each UAV utilizes a cognitive map regarding a search region as a knowledge base. Based on the information newly collected about the environment, the map is dynamically updated and used to route UAVs. The objective of the search is to minimize the total uncertainty in the region. Sujit and Ghose [13] present a search problem that incorporates an endurance time constraint on UAVs, and propose an algorithm that utilizes the k-shortest path algorithm with the objective of minimization of uncertainty in a region. The other works that employ a cognitive map and use a measure of uncertainty for the construction of search path can be found in the literature [14–19].

While the search algorithms designed to minimize uncertainty yields good performance in stationary target search problems, the algorithms' performance may deteriorate in dynamic environments where targets are assumed to move. Polycarpou et al. [12] state that the algorithm can be adjusted by applying a decay factor on uncertainty values in the case of dynamic environments. However, this model does not fully address the dynamic environment as reported in [20] since the uncertainty measure will continuously increase at all points that are not under direct surveillance. The other approaches to solve a search problem for moving targets include probability-map based search and exhaustive search such as in [21–23].

Other than search path planning, task allocation over targets is also of great interest to a team of multiple agents. Some researchers have formulated the task allocation problem in optimization model and proposed solution approaches based on centralized control mechanism utilizing mixed integer linear programming, dynamic programming and genetic algorithms, which can be found in [24–28]. Though the centralized mechanisms have good advantages including guaranteeing optimal solution, decentralized approaches are more favored and common in dynamic and uncertain environments because of robustness, quick response to dynamic events, and little computation overhead. Lemaire et al. [29] present a distributed task allocation scheme based on contract net protocol for multiple UAVs. Gurfil [30] investigate the performance of a team of UAVs that performs a search and destroy mission utilizing auction based task allocation. Jin et al. [2] address a search and response problem where a heterogeneous team of UAVs performs various tasks including search, attack, and damage assessment. The authors propose a predictive assignment algorithm where the UAVs estimate probabilities for future tasks and incorporate these predictions into their decision making. Sujit et al. propose a negotiation based task allocation scheme for multiple UAVs that have communication constraints [31], and further develop the scheme using sequential auctions [32].

Most of the work mentioned above study either the problems that consider separately search and task allocation issues or the problems that integrates the two issues only for stationary targets. However, for a complex mission where multiple

UAVs are supposed to deal with moving targets, there is a need to develop an efficient task allocation mechanism combined with search planning in a distributed setting. In the following sections, we describe the problem in detail and develop a distributed search and task allocation scheme for the problem.

3 Problem Statement

We consider a team of UAVs deployed into a certain battlefield where the team is supposed to perform search and attack tasks for targets that move around in the battlefield. The UAVs are modeled as autonomous distributed agents that make their own decisions based on the knowledge about the environment. In the rest of the paper, we use the general term "agent" to represent a UAV.

3.1 Mission Space

The continuous battlefield is partitioned into a collection of identically-sized grid cells, $C = \{1, 2, \ldots, N_C\}$, as shown in Fig. 1. To describe the movements of agents and targets in this discretized environment, we define the movement period T as the time interval that discretizes the movements of agents and targets. In the model, agents and targets move among the cells at each discrete time step $t = kT$ for $k = 0, 1, 2, \ldots$

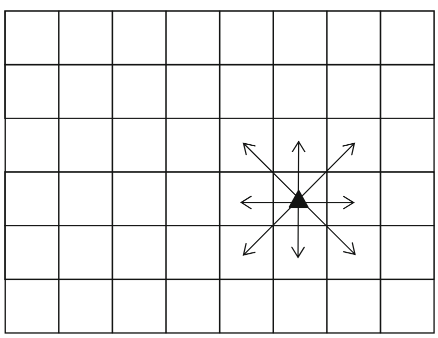

Fig. 1 Movement of agent in grid environment

In the grid environment, a cell represents a region where an agent expends its search effort or strikes a target detected. It is assumed that a cell is large enough for an agent to maneuver inside it, performing a search or attack task. At each time step, an agent determines a cell for search according to a search mechanism, and a sequence of cells that the agent chooses constitutes a search path of the agent.

3.2 Targets

There are N_T moving targets in the environment. The targets are probabilistically located at time $t = 0$ and move among cells according to a Markov process, each occupying one cell at each time interval. It is assumed that the initial location probabilities and transition probabilities of each target are known and independent of others. For simplicity, we assume that the targets move according to a homogeneous Markov chain Λ.

The targets are of various types and each target is allotted a value based on its priority or importance level. A target value is time-dependent, typically discounted in time, and is specified by its value function,

$$V_j(t) = V_j^0 \cdot \varphi_j(t) \tag{1}$$

where V_j^0 is a positive constant representing an initial value assigned to target j and $\varphi_j(t) \in [0,1]$ is a function defined to be non-increasing over time. Each of these targets requires some amount of munition resources for destruction and the amount may vary depending on the type of munition that an agent carries.

3.3 Agents

A team of agents consists of N_A heterogeneous agents. Each agent moves autonomously through the environment executing tasks such as search and attack. The agents carry limited fuel resources and are assumed to have no refueling during mission. It is assumed that global communication between the agents is possible and there is no interruption or delay in communication.

3.3.1 Movement of Agent

During search, an agent moves from one cell to another cell or stays in a cell at each time step. The available location at the next time step is constrained to one of adjacent cells to the agent's current location (See Fig. 1). When conducting a search task in a cell, an agent moves around searching for targets. We do not explicitly model the internal dynamics of the individual UAV (when maneuvering in a cell) and hence concentrate on the development of mechanisms for search planning and task allocation. However, when an agent needs to move to attack a target in other location, the agent flies directly from the current location to a cell where it is supposed to perform the attack task. We assume that the agent moves in its optimal speed, v_{attack}, so as to carry out the attack task in a minimum time.

A target may change its location while an agent is heading to attack the target, and thus the agent initially computes its expected travel time based on the location where the target is spotted. The distance between two locations is measured in Euclidean distance from the center of one cell to the center of the other cell as illustrated in Fig. 2. Then the expected travel time that agent i in cell c_i takes to arrive cellc_j where target j is currently located is computed as

$$\Delta t_{ij} = \frac{dist(c_i, c_j)}{v_{\text{attack}}} = \frac{\sqrt{\Delta x^2(c_i, c_j) + \Delta y^2(c_i, c_j)}}{v_{\text{attack}}} \tag{2}$$

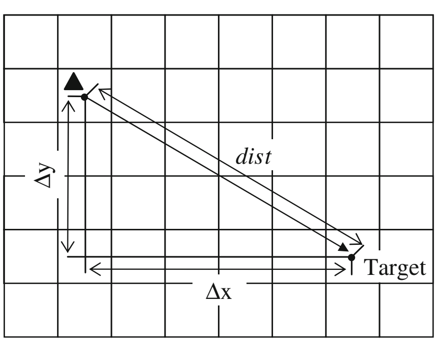

Fig. 2 Euclidean path of agent for attacking a target

In the model, we define the response time for agent i to attack target j as the minimum number of time steps to arrive to the current location of the target, which is computed as

$$\tau_{ij} = \left\lceil \frac{\Delta t_{ij}}{T} \right\rceil \tag{3}$$

where $\lceil r \rceil$ denotes the next higher integer of real number r and T represents the time interval. While moving to attack a target, an agent adjusts its path as it obtains new information about the target's location and searches no cells that it moves through in order to reach the target in a minimum time.

3.3.2 Capabilities of Agent

An agent is equipped with a sensing device, with which the agent is assumed to be able to detect a target with a certain probability provided that the agent and the target are located in the same cell. The detection capability of agents might be different depending on the type of the agent and the search condition of a region. The detection capability of agent i is defined by a detection function,

$$\beta_i(c, t) = 1 - e^{-\alpha(c,t)} \tag{4}$$

where $\alpha(c, t) \geq 0$ represents the search effectiveness of cell c at time t, which is normally used in search problems. We assume that, once an agent detects a target, the agent can identify physical properties of the target such as type, value and munition resources requirement as well.

In order to destroy a target that has been identified, the target needs to be serviced by an agent that holds proper attack capability over the target. An agent carries some munition resources limited in quantity and depletes with usage. While engaged with a target, an agent delivers some amount of munitions to the target and destroys the target with kill probability, which is determined based on the effectiveness of munitions the agent carries. The kill probability of agent i over target j is defined as

$$\delta_{ij}\left(\mu_{ij}(t)\right) = \text{Probability}\left(\text{attack destorys target } j \,\middle|\, \mu_{ij}(t)\right) \tag{5}$$

where $\mu_{ij}(t)$ represents the amount of available munition resources at time t that agent i can use to destroy target j. The attack capability of agent i is specified by its kill probability to the target and defined as follows.

$$\gamma_{ij}(t) = \begin{cases} \delta_{ij}\left(\mu_{ij}(t)\right), & \text{if agent } \delta_{ij}\left(\mu_{ij}(t)\right) \geq \delta_{\min} \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

An agent is not eligible to attack target j if its kill probability is less than some positive number $\delta_{\min}$ or it has no sufficient amount of munition resources required to attack the target. Following the attack task on a target, Battle Damage Assessment (BDA) is instantly performed by the attacking agent and assumed to be precise. If the attack task fails, the agent is required to re-attack the target until the target is successfully destroyed.

3.4 The Objective of Mission

During the mission individual agent collects information about targets and environment, and shares this information via communication with other agents. Utilizing this information, a team of agent collaborates to search for and destroy targets. When an agent successfully destroys a target, the agent is awarded a reward corresponding to the value of the target at that time. Then the objective of the mission for the team is to maximize the total rewards attained by successfully destroying targets in a given mission horizon. To accomplish this goal, the team needs to cooperatively work in the efficient search planning and effective task allocation mechanism.

4 The Agent Information Base

In order to effectively search for targets, agents must keep the state information of the environment in terms of targets location. To do this, two information maps are used as knowledge base for planning search route.

4.1 Target Location Probability Map

Each agent has a target location probability map $Q(t = kT)$ that contains probability information regarding targets location. In the map, each cell i has a value, termed the target location probability, $q(i,\ j, t) \in [0,1]$ representing the probability that target j is present in the cell at time t. The location probability map is constructed with the initial target location probability distribution of each target, $Q(t = 0) = \{q(i,\ j, 0)\}$, known in advance and is updated at each time step according to a Markov process Λ as new target location information is obtained.

In the absence of search, the target location distribution evolves according to the formula,

$$Q(t = kT + T) = Q(t = kT) \cdot \Lambda \tag{7}$$

where $\Lambda = \{\lambda_{ij} | i,\ j \in C\}$ represents the target transition probability matrix. However, incorporated with the likelihood of overlooking targets in a search, the probability of target j being present in cell i is computed as

$$q(i,\ j, t + T) = \sum_{k=adj(i)} \lambda_{ki} \cdot q(k,\ j, t) \cdot \psi_a(k, t) \tag{8}$$

where $adj(i)$ represents a set of adjacent cells to cell i including the cell itself and $\psi_a(k,t)$ is the probability that agent a overlooks targets in cell $k \in adj(i)$, which is defined as

$$\psi_a(k, t) = \begin{cases} 1 - \beta_a(k, t), & \text{if agent } a \text{ searches cell } k \\ 1, & \text{at time t otherwise} \end{cases} \tag{9}$$

Agents cooperatively build a target location probability map, utilizing new information each agent gathers during each time interval. As an agent carries out a task in a cell, it obtains some information about targets, e.g. targets detected, undetected, or destroyed. The agent propagates this information to other agents via communication and updates the target location probability map according to the formula (7) and (8). For example, if a target is detected in a cell, the location probability of the target is set to 1.0 in the cell and 0 for all other cells in the map. Once a target is destroyed, the target's location information is removed from the location probability map.

Since all information is shared among agents through global communication, every agent keeps the same target location probability map at any time. Each agent utilizes the target location probability map and a certainty map detailed in the following section, as a guide for autonomous decision making in its own search path planning.

4.2 Certainty Map

In the target location probability map, each cell contains estimated probabilities of targets' existence. However, in general, the probabilities may not precisely reflect real probabilities of targets' location due to inherent uncertainty in the probabilities. For example, if agents have detected no target for a long period of time, the probabilities of targets' existence, which have been developed from Eqs. 7 and 8 over the time, come out with only a small value. It implies that the location probability distribution is un-normalized through the entire environment and thus there may be a high uncertainty in the probabilities. Therefore, using only the target location probability map may not provide sufficiently good information that account for targets location. For this reason, in our model another measure, called certainty, is used to capture the moment information describing ambiguity in the location probabilities as used in [14–16].

We define a certainty variable, $h(i,t) \in [0,1]$, for each cell i, which quantifies information deficiency about target location probabilities in the cell. This certainty value corresponds to the degree to which the cell has been searched and drives agents to explore un-searched region. The certainty value $h(i,t) = 0$ implies that the cell has not been searched for a long time and thus the location probabilities in the cell is quite uncertain. As the cell is searched repeatedly, its certainty value approaches 1.0. The map that stores this information is called the certainty map, $H(t = kT)$, which begins with an initial certainty value in each cell and is updated at each time step as the cells are searched. Each time an agent visits a cell and performs a search, the certainty value of the cell changes according to the rule described in [15, 16];

$$h(i, t + T) = h(i, t) + 0.5\,(1 - h(i, t)) \tag{10}$$

However, in a dynamic environment where targets move, if no search is performed in a cell, the cell's certainty level decays with time in cope with a changing environment so that cells scanned in recent search have more weight on the certainty level. In our model, the certainty value of a cell with no search decreases using a discount factor $\varepsilon \in [0,1]$ as

$$h(i, t + T) = \varepsilon \cdot h(i, t) \tag{11}$$

This update rule is a simple way for agents to track the number of searches recently conducted on each cell and capture the notion of uncertainty in location probability information of targets. The certainty map constantly changes over a mission horizon and is shared among all agents through communication.

5 Proposed Approach

The approach proposed in this paper utilizes a probabilistic decision making protocol based on the response threshold model described in [3, 4]. In the response threshold model, each agent is given a set of thresholds and each task is assigned a stimulus. The threshold represents the preference or specialization of an agent toward a task and can be updated in regard with the change of state. The stimulus of a task indicates the demand or value of the task itself and is used to attract agents' attention. A probability function is defined using the threshold and the stimulus, by which an agent determines the probability that it takes a task.

In our model, tasks include a set of search tasks, each for searching an individual cell, and a set of attack tasks, each for attacking an individual target detected. To each set of search and attack tasks, the response thresholds of individual agent i are given as $\Theta_i^S = \{\theta_{i1}^S(t), \cdots, \theta_{i,N_C}^S(t)\}$ and $\Theta_i^A = \{\theta_{i1}^A(t), \cdots, \theta_{i,N_T}^A(t)\}$, respectively. The stimulus of a task is determined based on total expected values of targets. While the response threshold of agents and the stimulus of tasks are defined differently for the search and attack tasks as detailed in the following sub-sections, agents are designed to operate in a similar manner; the agent selects probabilistically its action by evaluating task according to a probability function. Though there are a couple of different forms of the probability function used in the literature, we adopt a function in3, i.e., given threshold θ_{ij} and stimulus S_j, agent i will take task j with probability:

$$P(S_j, \theta_{ij}) = \frac{S_j^2}{S_j^2 + a \cdot \theta_{ij}^2 + \Delta\tau_{ij}^{2 \cdot b}} \tag{12}$$

where $\Delta\tau_{ij}$ is the time period taken until agent i starts task j, and a and b are parameters. The lower an individual's threshold or the higher a task's stimulus, the more likely the agent accepts the task.

The task allocation framework we propose consists of two mechanisms; search planning mechanism for search tasks and target assignment mechanism for attack tasks. At each time step, every agent is introduced search tasks, each of which is to search one of adjacent cells. An agent selects one search task through the search planning mechanism. During search, attack tasks would be introduced if targets are detected. In our model, attack tasks have higher priority over search tasks. An agent that detects a target runs the target assignment mechanism through which an attack task over the target is assigned. The overall flow of the task allocation framework is shown in Fig. 3.

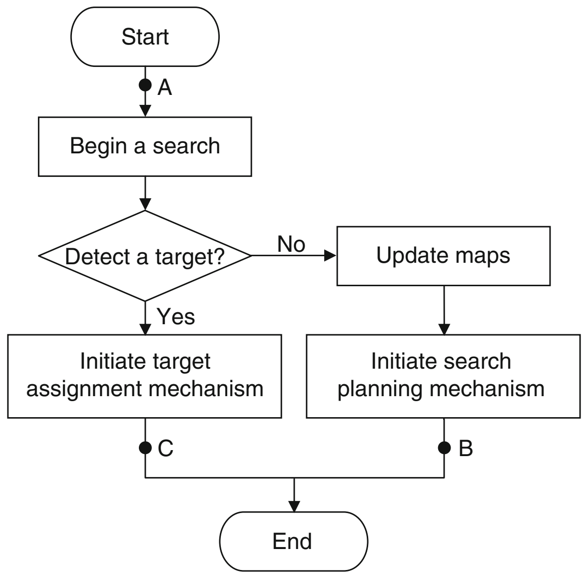

Fig. 3 Task allocation framework flowchart

5.1 Search Planning Mechanism

An agent performs a search (as a default task) if no target is detected at the time or the agent is not eligible to any of attack tasks announced. At each time step, an agent selects one search task from available search tasks for the next time interval. The response threshold to and the stimulus of an individual search task are defined using the environment state information at the time such as target location probabilities in the location probability map, $Q(t)$, and certainty values in the certainty map, $H(t)$.

In the mechanism, the response threshold of agent i to a search task in cell j, is given by the cell's certainty value$h(j,t)$ at the time and bounded in $[0,\theta_{\max}]$, which is defined as:

$$\theta_{ij}^S(t) = \theta_{\max} \cdot \left(1 - e^{-\omega \cdot h(j,t)}\right) \tag{13}$$

where $\omega > 0$ is a parameter. Note that agents have all common response thresholds to search tasks since the thresholds are determined only with cells' certainty value. The response thresholds are updated every time step as certainty value of cells changes. A stimulus of a search task in cell j is defined based on potential possibilities to detect targets in the cell and values of the targets, which is given by:

$$S_j^S(t) = \sum_{k \in N_T} q(j,k,t) \cdot \beta_i(j,t) \cdot V_k(t) \tag{14}$$

In our model, no delay time is required to transfer to the next search task, which means $\Delta\tau_{ij} = 0$. Then, according to the Eq. 12, the probability that agent i accepts a search task for cell j at time t is computed as:

$$\widetilde{P}_{ij}^S(t) = \frac{\left(S_j^S(t)\right)^2}{\left(S_j^S(t)\right)^2 + a \cdot \left(\theta_{ij}^S(t)\right)^2} \tag{15}$$

We interpret this probability as the preference level of an agent to a particular search task. Since an agent is given multiple search tasks, the agent needs to pick only one among them for the next time search. Thus a real probability for an agent to select a search task is given by a normalized probability using the preference levels as:

$$P_{ij}^S(t) = \frac{\widetilde{P}_{ij}^S(t)}{\sum_{k \in adj(c)} \widetilde{P}_{ik}^S(t)} \tag{16}$$

According to the function (15), the lower response threshold or the greater stimulus is given, the higher preference level is imposed to a search task in a cell, which means the cell is more likely to be visited. In this way, agents are driven to explore cells not only with high target location probabilities but also with low certainty value.

However, there might be a case where more than one agent competes for the same cell to search. In that case, a conflict resolution rule is applied to avoid overlap in search efforts; the cell is given to an agent with higher preference level and the other one takes other cell with the second highest preference level. Since there are no more than one UAV in one cell simultaneously, collision-free operation (flight) is guaranteed when searching in a cell. Figure 4 shows the

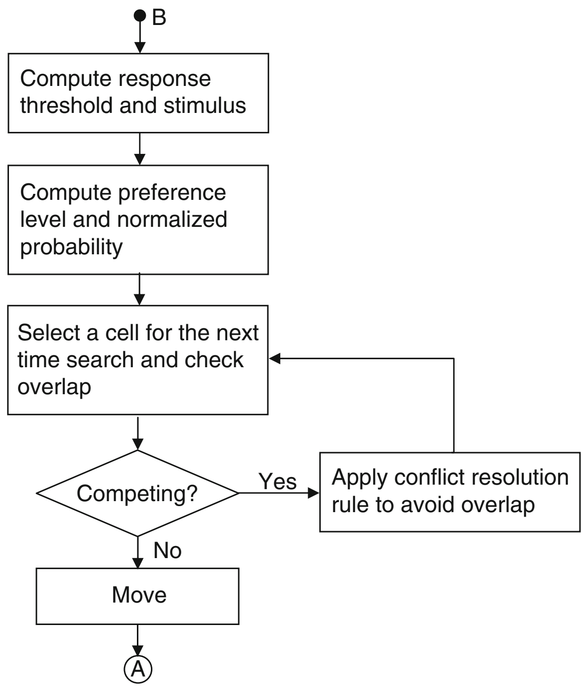

Fig. 4 Search planning mechanism flowchart

flowchart that details the process of the search planning mechanism.

5.2 Target Assignment Mechanism

When an agent senses a target, it initiates target assignment mechanism to select a best suitable agent to perform the attack task over the target. If an agent detects multiple targets, it proceeds with target assignment process in parallel for each target. Target assignment mechanism in our model utilizes auction-like protocol. The process starts with announcement of an attack task along with information associated with a target such as type, location, munitions requirement, stimulus, etc. Each other agent that receives the attack task information checks its eligibility, i.e., whether or not the agent holds a proper attack capability with sufficient munition resources, and if so, respond to the task with a probability given from function (12).

An agent's response threshold to attack task is given based on the agent's attack capability. If an agent has high kill probability over a target, it gets low response threshold to a corresponding attack task. Therefore, the response threshold is defined to be inversely proportional to kill probability as follows:

$$\theta_{ij}^{A}(t) = c - d \cdot \gamma_{ij}(t) \tag{17}$$

where c and d are positive constants and θ_{ij}^{A} is bounded in $[0,\theta_{\max}]$. If the agent has no attack capability on a target, the response threshold to the attack task is set to infinity. A stimulus of an attack task is simply determined as a corresponding target's value at the time, i.e.,

$$S_{j}^{A}(t) = V_{j}(t) \tag{18}$$

Once the agent decides to respond, it sends a response message including some information such as estimated response time (τ_{ij}), kill probability, and expected rewards gain to the auctioneering agent. The expected reward of agent i for destroying target j is computed as

$$E_{ij}\left(V_{j}(t+\tau_{ij})\right) = \gamma_{ij}(t) \cdot V_{j}(t+\tau_{ij}) \tag{19}$$

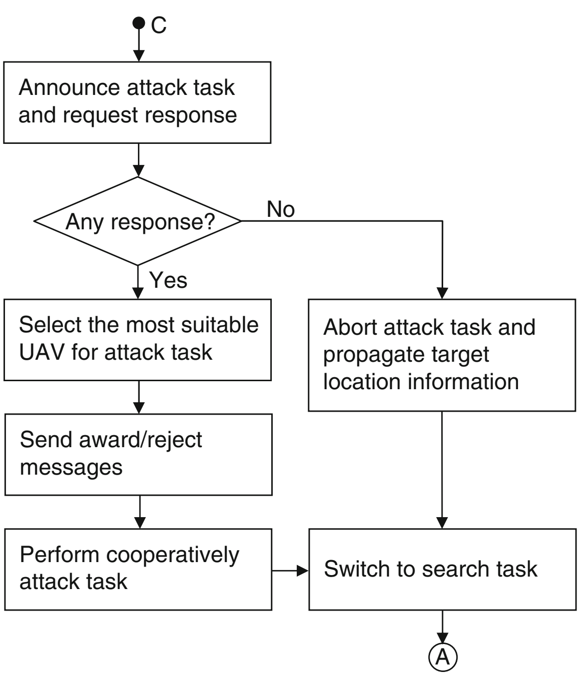

Fig. 5 Target assignment mechanism flowchart

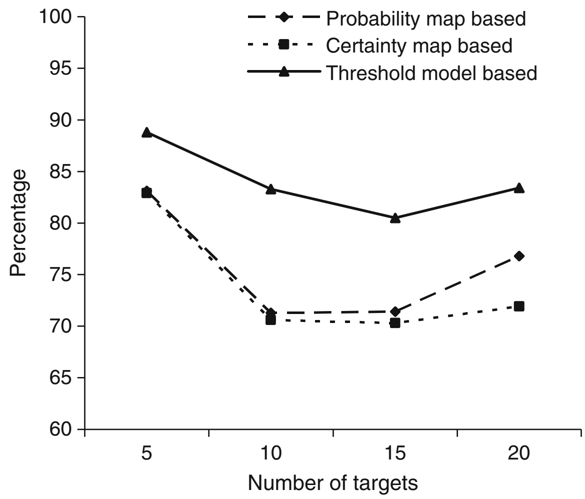

Fig. 6 The performance of each method by varying the number of targets when 5 UAVs and $\varepsilon = 0.90$ are used

An agent may have a schedule of attack tasks previously assigned. Then its response time is the total time to complete all the attack tasks in the schedule, plus estimated time to reach the announced target after finishing the last task in the schedule. When multiple attack tasks are concurrently announced, an agent performs the same decision making process to select an attack task as does in the search task selection, i.e., evaluates a response probability for each of attack tasks, and selects a task by a normalized probability.

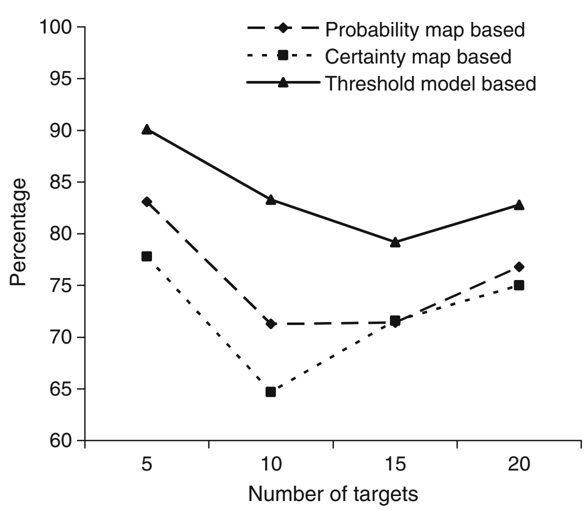

Fig. 7 The performance of each method by varying the number of targets when 5 UAVs and $\varepsilon = 0.95$ are used

Table 1 The capability of UAV

	Detection	Kill				
		Type A Target	Type B Target	Type C Target	Type D Target	Type E Target
Type I UAV	0.9	0.5	0.2	0.9	0.8	0.85
Type II UAV	0.85	0.7	0.95	0.85	0.3	0.4
Type III UAV	0.95	0.9	0.75	0.7	0.75	0.7

After an agent collects all responses from eligible agents, the agent chooses the best one that can produce the largest expected reward. The agent sends a reject message to other agents that are not selected. In the event that an agent does not get any response from other agents, then the agent aborts the attack task and switches to perform a search task. However, even in this case, the agent propagates location information about the detected target so that other agents use this information to update a target location probability map and build a direction in future search plan.

Agents cooperatively perform an attack task. Once an agent assigns an attack task to other agent, the agent chases a target until the other agent reaches the target to attack. While chasing the target, the agent communicates the target's location information with the other agent. This cooperative attack task execution helps an attacking agent navigate to the final location where the attack task needs to be conducted. The agents are released from the attack task and begin other tasks when the task is completed. The overall process of the target assignment mechanism is illustrated in Fig. 5.

6 Simulation Results

In this section we demonstrate the performance of the proposed approach compared to two other methods using a simulation environment; target location probability map based [11] and uncertainty map based search and task allocation method [12]. Since these methods are designed only to address search path planning, we modify the methods to include deterministic attack task allocation scheme for comparison with our approach. In the deterministic task allocation the two methods use, an agent must respond to an announced attack task if the agent is eligible to the task. Each method is described as follows.

Target location probability map based search and task allocation This approach is a somewhat greedy method that utilizes only a target location probability map. At each time step an agent selects one cell for search that has the highest location probability weighted by targets' value. Upon detecting a target, an agent immediately performs an attack task over the target if it has proper attack capability. Otherwise, the attack task is deterministically assigned to the other agent that is expected to yield the best reward by destroying the target.

Certainty map based search and task allocation In this approach, an agent uses only a certainty map for routing its search path, that is, an agent moves to a cell with the lowest certainty level for the next time search. The certainty value of a cell dynamically changes as described in the previous section.

Table 2 The average number of targets destroyed, total cumulative reward and mission completion time of each method when $\varepsilon = 0.90$ is applied in case study 1

No. of targets	Probability map based method			Certainty map based method			Response threshold model based		
	Targets destroyed	Rewards gain	Completion time	Targets destroyed	Rewards gain	Completion time	Targets destroyed	Rewards gain	Completion time
5	4.4	299.0	90.3	4.5	298.5	88.8	4.7	319.5	74.7
10	7.6	513.5	120.0	7.8	508.0	114.2	8.6	600.0	114.8
15	11.3	771.0	120.0	11.3	759.0	120.0	12.6	869.0	120.0
20	16.4	1106.5	120.0	15.4	1035.0	120.0	17.6	1200.5	116.9

Table 3 The average number of targets destroyed, total cumulative rewards and mission completion time of each method when $\varepsilon = 0.95$ is applied in case study 1

No. of targets	Probability map based method			Certainty map based method			Response threshold model based		
	Targets destroyed	Rewards gain	Completion time	Targets destroyed	Rewards gain	Completion time	Targets destroyed	Rewards gain	Completion time
5	4.4	299.0	90.3	4.3	280.0	97.0	4.8	324.5	65.2
10	7.6	513.5	120.0	7.1	466.0	119.4	8.7	599.5	115.0
15	11.3	771.0	120.0	11.5	773.5	120.0	12.4	855.0	116.5
20	16.4	1106.5	120.0	16.3	1080.5	120.0	17.5	1192.0	120.0

The attack task allocation process is performed in the same manner as the target location probability map based method.

6.1 Simulation Scenario

The simulation is conducted with a simulated UAV team that performs a search and destroy mission in a battlefield. A battlespace is represented by 10×10 grid cells with the size of 2 km for each side. In the environment there are five types of targets that move around. The initial value of target is given 100, 90, 70, 60 and 40 for type A, B, C, D, and E, respectively. The value of a target decreases according to the function

$$V_j(t) = V_j^0 \cdot \left(\frac{1}{1 + \eta \cdot t}\right) \tag{20}$$

where η is set to 0.03. We consider three types of UAVs, of which munition resources is given between 20 to 60 units. The munitions usage of UAV to attack a target is varied between 10 to 20 units. For simplicity, the detection and kill probability of UAV to each type of target are set as shown in Table 1. The speed of UAV for moving to attack is identical and constant, which is assumed to be 300 km/hr. The movement period T and the mission horizon are set to one minute and two hours, respectively.

For each simulation run, the initial probability distribution of target location is randomly generated and each target is located in one of cells of which $q(i, j, 0) > 0$. The certainty value of each cell is initially set to 1.0. The parameters of the system are set as follows: $\delta_{\min} = 0.7$; $a = 0.01$; $b = 1.0$; $\theta_{\max} = 100$; $\omega = 3.0$; $c = 333.33$; $d = 333.33$. The algorithms are coded in MATLAB, and the

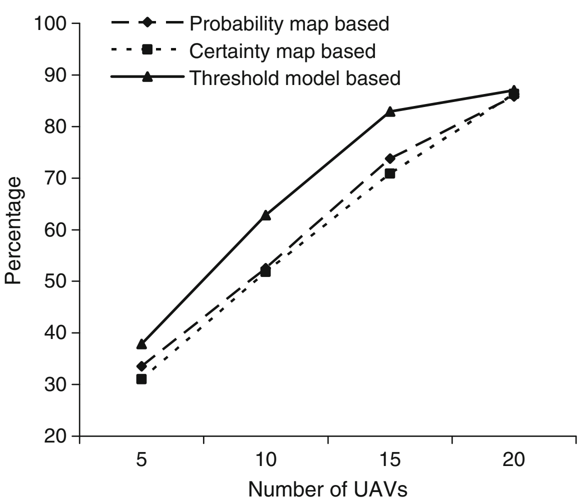

Fig. 8 The performance of each method by varying the number of UAVs when 15 targets and $\varepsilon = 0.90$ are used

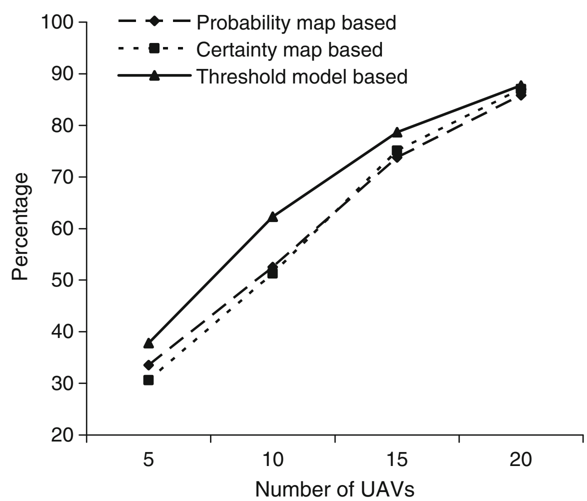

Fig. 9 The performance of each method by varying the number of UAVs when 15 targets and $\varepsilon = 0.95$ are used

Table 4 The average number of targets destroyed, total cumulative rewards and mission completion time of each method when $\varepsilon = 0.90$ is applied in case study 2

No. of UAVS	Probability map based method			Certainty map based method			Response threshold model based		
	Targets destroyed	Rewards gain	Completion time	Targets destroyed	Rewards gain	Completion time	Targets destroyed	Rewards gain	Completion time
3	5.3	362.0	120.0	4.8	335.0	120.0	5.9	409.0	120.0
5	8.2	567.0	120.0	8.6	559.5	120.0	9.6	678.5	120.0
7	11.8	797.0	120.0	11.4	765.5	120.0	13.2	895.5	116.7
9	13.5	926.3	114.0	13.5	931.5	102.2	13.6	939.5	105.3

simulation results are the average of 50 runs for each case.

6.2 Results and Discussion

Wetest the response threshold based method comparing to the other algorithms by varying simulation environment. The performance is measured with total cumulative rewards that a UAV team collects by successfully destroying targets over a fixed time window. Then the performance is represented in percentage obtained by dividing the total rewards gain by total sum of target initial values.

Case study 1 In the first set of simulation, a UAV team composed of 5 UAVs (2 of each type A and B, and 1 of type C) is used given different number of targets varied from 5 to 20. Figures 6 and 7 demonstrate the performance of each method with different discount factor values ($\varepsilon = 0.90$ and $\varepsilon = 0.95$). As seen in the figures, the response threshold model based method performs the best in all instances, achieving up to 12.1 % and 12.7 % more rewards than probability map and certainty map based method respectively when $\varepsilon = 0.90$, and 11.9 % and 18.5 % more than probability map and certainty map based method respectively when $\varepsilon = 0.95$. We measure the average number of target destroyed, the total cumulative rewards and mission completion time as listed in Tables 2 and 3. Obviously, as more targets are injected into the system, each method obtains more rewards by detecting and destroying more targets. In addition, the response threshold model based approach completes mission in shorter time since a UAV team can have better chances to find targets in the proposed approach. Utilizing state information including targets location probabilities and certainty variables, our approach reduces uncertainty in search environment and provides flexibility for routing a search path. When assigning attack tasks, the proposed approach determines the most suitable UAV considering its attack capability and response time, and hence produces more rewards.

Case study 2 The second set of simulation compares the performance by varying the number of UAVs given 15 targets. Figures 8 and 9 illustrate the performance of each method when discount factor value $\varepsilon = 0.90$ and $\varepsilon = 0.95$ is applied, which show that the response threshold model based method outperforms the other two methods in both cases. Our method attains at most 10.3 % more rewards that probability map based method

Table 5 The average number of targets destroyed, total cumulative rewards and mission completion time of each method when $\varepsilon = 0.95$ is applied in case study 2

No. of UAVS	Probability map based method			Certainty map based method			Response threshold model based		
	Targets destroyed	Rewards gain	Completion time	Targets destroyed	Rewards gain	Completion time	Targets destroyed	Rewards gain	Completion time
3	5.3	362.0	120.0	4.9	331.0	120.0	5.7	408.5	120.0
5	8.2	567.0	120.0	8.1	553.5	120.0	9.6	672.5	120.0
7	11.8	797.0	120.0	12.0	811.0	120.0	12.3	849.5	120.0
9	13.5	926.3	114.0	13.7	940.0	111.3	13.8	947.0	105.1

when $\varepsilon = 0.90$ and 11.0 % more than certainty map based method when $\varepsilon = 0.95$. Tables 4 and 5 summarize the average number of targets destroyed, the total cumulative rewards and mission completion time of each method. The response threshold model based method detects and destroys more targets in a given time duration and completes faster a mission. This confirms that the proposed approach can more efficiently explore the environment and more effectively allocates tasks. Notice that the difference in performance is not so significant when a small or large number of UAVs is used. This is because a small size UAV team has a limited amount of munition resources in total, which is not sufficient to handle all targets. In contrast, a large size UAV team has many available UAVs with sufficient munition resources so that it can immediately react to a detected target. This suggests that the response threshold based method would become more superior if a proper number of UAVs are used.

Case study 3 The third set of simulation investigates the effect of certainty discount factor on the performance of response threshold model based method. In the first simulation, 5 UAVs given 10 targets and 7 UAVs given 15 targets are tried with various discount factor values from 0.8 to 1.0. Figures 10 and 11 illustrate the performance of each method. Observe that the performance of response threshold model based method seems not greatly influenced by discount factor value while the performance of certainty map based method fluctuates depending on discount factor value. This is because the response threshold model based method more flexibly explores the environment by probabilistically selecting a search cell whereas the certainty map based method deterministically chooses a search cell with the highest uncertainty. It suggests that our method works

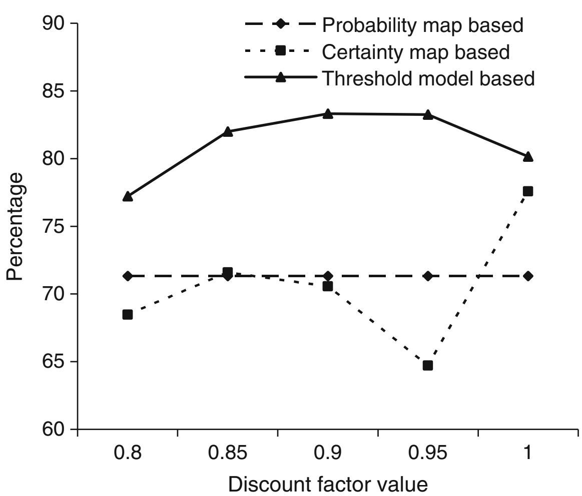

Fig. 10 The comparison of performances by varying discount factor value when 5 UAVs and 10 targets are used

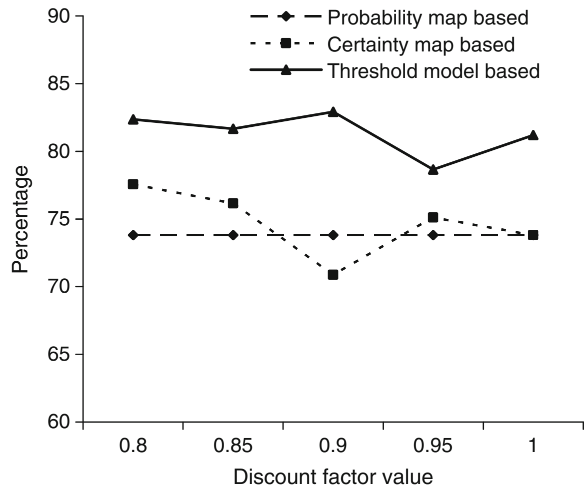

Fig. 11 The comparison of performances by varying discount factor value when 7 UAVs and 15 targets are used

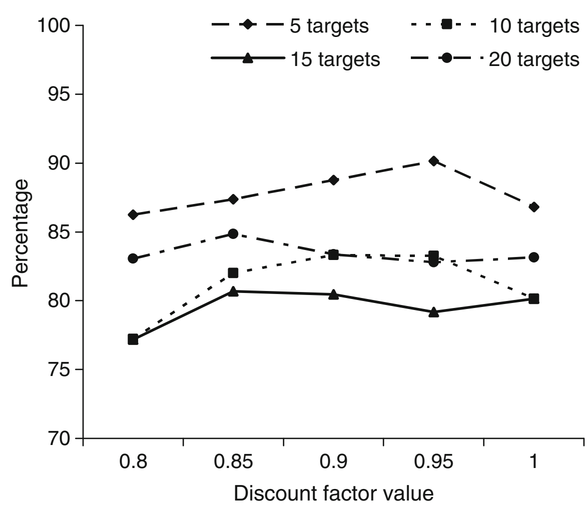

Fig. 12 The performance of the response threshold model based method with various number of targets and discount factor values when 5 UAVs are used

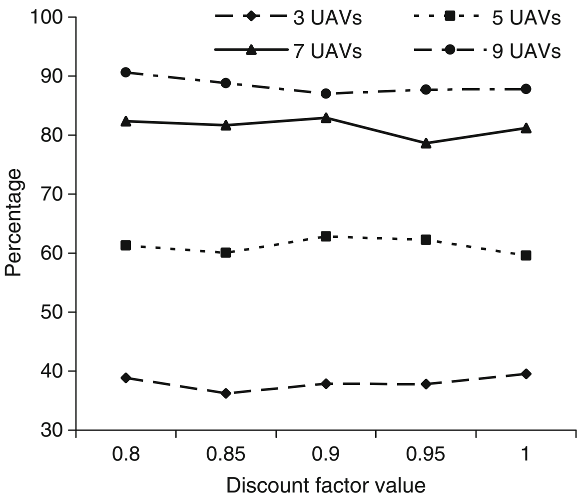

Fig. 13 The performance of the response threshold model based method with various number of UAVs and discount factor values when 15 targets are given

well in various situations yielding a stable performance. To verify this, the second simulation is conducted by varying the number of targets and UAVs. The results summarized in Figs. 12 and 13 show that the difference in performance is within only 6.1 %. This confirms that the proposed approach performs in a stable way better than the certainty map based method.

7 Conclusion

This paper presents a distributed search planning and task allocation approach for a heterogeneous UAV team that performs a search and destroy mission for moving targets. In the model, the movement of targets is described by a Markov process based on which target location probability information is developed. The certainty is used as a measure to account for ambiguity in the target location probability information. Using this environment state information, the UAV team cooperatively builds a target location probability map and a certainty map, which constitute a knowledge base for planning search route and task allocation. In the proposed approach, probabilistic decision making scheme based on response threshold model is developed to provide flexibility that enables the UAV team to efficiently explore an environment and carry out tasks. The proposed approach has been evaluated in simulation experiments compared to deterministic methods. The results clearly show that the proposed approach outperforms the other methods in various conditions. Future research will examine some practical issues such as targets' threats, communication constraints and failures, UAV malfunctions and shoot-downs, and task failures for the extension of the problem. Those issues could significantly affect the system performance in a real battlefield and hence should be addressed in a model. We will also investigate the impact of parameters of the model on the system and develop techniques to tune the parameters according to the condition of environment.

References

1. Unmanned Aircraft Systems Roadmap 2005–2030. Office of Secretary of Defense (2005)
2. Jin, Y., Liao, Y., Minai, A., Polycarpou, M.: Balancing search and target response in cooperative Unmanned Aerial Vehicle (UAV) team. IEEE Trans. Syst. Man Cybern. B Cybern. **36**, 571–587 (2006)
3. Campos, M., Bonabeau, E., Theraulaz, G., Deneubourg, J.-L.: Dynamic scheduling and division of labor in social insects. Adapt. Behav. **8**, 83–92 (2001)
4. Nouyan, S., Ghizzioli, R., Birattari, M., Dorigo, M.: An insect-based algorithm for the dynamic task allocation problem. Künstlich Intell. **4**, 25–31 (2005)
5. Benkoski, S., Monticino, M., Weisinger, J.: A survey of the search theory literature. Nav. Res. Logist. **38**, 469–494 (1991)
6. Stone D.: Theory of Optimal Search, 2nd edn. Operation Research Society of America, Arlington (1989)
7. Washburn, A.: On a search for a moving target. Nav. Res. Logist. **27**, 315–322 (1980)
8. Brown, S.: Optimal search for a moving target in discrete time and space. Oper. Res. **28**, 1275–1289 (1980)
9. Stewart, T.: Search for a moving target when searcher motion is restricted. Comput. Oper. Res. **6**, 129–140 (1979)
10. Washburn, A.: Search for a moving target: the FAB algorithm. Oper. Res. **31**, 739–751 (1983)
11. Grundel, D.: Searching for a moving target: optimal path panning. In: Proceedings of the IEEE International Conference on Networking, Sensing, and Control, pp. 867–872. Los Alamitos, CA (2005)
12. Polycarpou, M., Yang, Y., Passino, K.: A cooperative search framework for distributed agents. In: Proceedings of the 2001 IEEE International Symposium on Intelligent Control, pp. 1–6. Mexico City, Mexico (2001)

13. Sujit, P., Ghose, D.: Search using multiple UAVs with flight time constraints. IEEE Trans. Aerosp. Electron. Syst. **40**, 491–508 (2004)
14. Yang, Y., Minai, A., Polycarpou, M.: Evidential map-building approaches for multi-UAV cooperative search. In: Proceedings of American Control Conference, pp. 116–121. Portland, OR (2005)
15. Jin, Y., Minai, A., Polycarpou, M.: Cooperative real-time search and task allocation in UAV teams. In: Proceedings of IEEE International Conference on Decision and Control, pp. 7–12. Maui, Hawaii (2003)
16. Flint, M., Fernandez-Gaucherand, E., Polycarpou, M.: Cooperative control for UAV's searching risky environments for targets. In: Proceedings of IEEE International Conference on Decision and Control, pp. 3567–3572. Maui, Hawaii (2003)
17. Yang, Y., Minai, A., Polycarpou, M.: Decentralized cooperative search by networked UAVs in an uncertain environment. In: Proceedings of American Control Conference, pp. 5558–5563. Boston, Massachusetts (2004)
18. Flint, M., Polycarpou, M., Fernandez-Gaucherand, E.: Cooperative control for multiple autonomous UAV's searching for targets. In: Proceedings of IEEE International Conference on Decision and Control, pp. 2823–2828. Las Vegas, Nevada (2002)
19. Sujit, P., Sinha, A., Ghose, D.: Team, game, and negotiation based intelligent autonomous UAV task allocation for wide area application. In: Chahl, J. S., Jain, L. C., Mizutani, A., Sato-Ilic, M. (eds.) Innovations in Intelligent Machines (1), pp. 39–75. Springer-Verlag, Heidelberg (2007)
20. Passino, K., Polycarpou, M., Jacques, D., Pachter, M., Liu, Y., Yang, Y., Flint, M., Barum, M.: Cooperative control for autonomous air vehicles. In: Cooperative Control and Optimization. Kluwer, Boston (2002)
21. Vincent, P., Rubin, I.: A framework and analysis for cooperative search using UAV swarms. In: Proceedings of the 19th Annual ACM Symposium on Applied Computing, pp. 79–86. Nicosia, Cyprus (2004)
22. Jun, M., D'Andrea, R.: Probability map building of uncertain dynamic environments with indistinguishable obstacles. In: Proceedings of American Control Conference, pp. 3417–3422. Denver, Colorado (2003)
23. Bertuccelli, L., How, J.: Search for dynamic targets with uncertain probability map. In: Proceedings of American Control Conference, pp. 737–742. Minneapolis, Minnesota (2006)
24. Shumacher, C., Chandler, P., Pachter, M., Pachter, L.: UAV task assignment with timing constraints via mixed-integer linear programming. In: Proceedings of AIAA 3rd the Unmanned Unlimited Systems Conference, pp. 232–252. Chicago, Illinois (2004)
25. Nygard, K., Chandler, P., Pachter, M.: Dynamic network flow optimization models for air vehicle resource allocation. In: Proceedings of the American Control Conference, pp. 1853–1858. Arlington, Texas (2001)
26. Shima, T., Rasmussen, S., Sparks, A., Passino, K.: Multiple task assignments for cooperating uninhabited aerial vehicles using genetic algorithms. Comput. Oper. Res. **33**, 3252–3269 (2006)
27. Arslan, G., Wolfe, J., Shamma, J. Speyer, J.: Optimal planning for autonomous air vehicle battle management. In: Proceedings of the IEEE Conference on Decision and Control, pp. 3782–3787. Las Vegas, Nevada (2002)
28. Shetty, V., Sudit, M., Nagi, R.: Priority-based assignment and routing of a fleet of unmanned combat aerial vehicles. Comput. Oper. Res. **35**, 1813–1828 (2008)
29. Lemaire, T., Alami, R., Lacroix, S.: A distributed task allocation scheme in multi-UAV context. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp. 1157–1162. New Orleans, Louisiana (2004)
30. Gurfil, P.: Evaluating UAV flock mission performance using Dudek's taxonomy. In: Proceedings of the American Control Conference, pp. 4679–4684. Portland, Oregon (2005)
31. Sujit, P., Sinha, A., Ghose, D.: Multiple UAV task allocation using negotiation. In: Proceedings of the International Conference on Autonomous Agents and Multiagent Systems, pp. 471–478. Hakodate, Hokkaido, Japan (2006)
32. Sujit, P., Beard, R.: Distributed sequential auctions for multiple UAV task allocation. In: Proceedings of the American Control Conference, pp. 3955–3960. New York City, New York (2007)