Distributed Probabilistic Search and Tracking of Agile Mobile Ground Targets Using a Network of Unmanned Aerial Vehicles

Chapter · November 2014

DOI: 10.1007/978-3-319-10807-0_14

3 authors:

Liang Sun
New Mexico State University

Stanley Baek
University of Michigan-Dearborn

D. Pack
University of Texas at San Antonio

Chapter 14
Distributed Probabilistic Search and Tracking of Agile Mobile Ground Targets Using a Network of Unmanned Aerial Vehicles

Liang Sun, Stanley Baek and Daniel Pack

Abstract As technologies in digital computation, sensing, wireless and wired communications, embedded systems, and micro-electro-mechanical systems continue to advance in the coming years, it is certain that we will see a variety of distributed sensor networks (DSNs) being deployed in an increasing number of systems such as power distribution systems, engineering structures and buildings, smart homes, environmental monitoring systems, biomedical systems, military systems, and others. In addition, unlike the traditional networks of sensors, the mobility afforded by autonomous systems, embedded systems, and humans who carry smart sensing devices will contribute in creating new and exciting future sensor networks. These future networks of sensors that take advantage of man-machine interactions will also introduce new applications yet unknown to us. In this paper, we present the origin and time line of DSN development, analyze the benefits and challenges of DSNs, and present a mobile sensor network in the form of an unmanned aerial vehicle (UAV) team using distributed mission area probability maps to search and track mobile ground targets. We propose a novel update strategy for the probability map used by UAVs to store probability information of dynamic target locations in the search area. Two update laws are developed to accommodate maps with different scales. Simulation results are used to demonstrate the validity of the proposed probability-map update strategy.

L. Sun (✉) · D. Pack
Department of Electrical and Computer Engineering,
The University of Texas at San Antonio, San Antonia, TX, USA
e-mail: liang.sun@utsa.edu

D. Pack
e-mail: daniel.pack@utsa.edu

S. Baek
Department of Electrical and Computer Engineering, The University of Michigan,
Dearborn, MI, USA
e-mail: stanbaek@umich.edu

P. Spagnolo et al. (eds.), *Human Behavior Understanding in Networked Sensing*,
DOI 10.1007/978-3-319-10807-0_14

14.1 Introduction

Sensors perceive real-world by capturing physical phenomena and converting them into electric signals. Combining these sensors, integrating them into a network of sensors, and applying its capability to applications, which benefit from synergistic sensor information, are the key functions of sensor networks. Today, a network of sensors, mostly static, detects catastrophic infrastructure failures, conserves precious natural resources, increases economic productivity, enhances security, and enables new applications such as context-aware systems and smart home technologies [1]. New sensor network technologies are also beginning to emerge with mobile sensor platforms, allowing smart and dynamic sensor placement in response to changing environment [2].

14.1.1 History of Distributed Sensor Networks

The origin of sensor networks can be traced back to defense applications developed during the Cold War. The Sound Surveillance System (SOSUS), a system of acoustic sensors at the bottom of the ocean, was deployed by the United States to detect and track submarines [3]. This system was later used by the National Oceanographic and Atmospheric Administration (NOAA) to monitor the migration patterns of whales and seismic signals indicating pending earthquakes [4, 5]. Sensor networks of air defense radars were also deployed during the Cold War era to defend England, the continental United States and Canada. Although research was focused on satisfying particular mission needs—in the case of SOSUS, acoustic signal processing and interpretation, tracking and fusion—these research produced some key technologies for modern sensor networks [6].

The research on distributed sensor networks was catalyzed around 1980 with the Distributed Sensor Network (DSN) project sponsored by the Defense Advanced Research Projects Agency (DARPA). By this time, the ARPAnet (the predecessor of the Internet) had been operational for a number of years, with about 200 hosts at universities and research institutions [6]. Focused on distributed computing, signal processing, and tracking, the DSNs postulated the possibility of spatially distributed sensing nodes designed to operate in a collaborative manner [3]. To support the distributed systems, in the late 1980s, researchers at Carnegie-Mellon University developed a communication oriented operating system kernel, Accent, to support transparent access and fault-tolerant behavior of a DSN [7, 8].

Recent advances in inexpensive low-power microprocessors, wireless networking, and micro-electro-mechanical systems (MEMS) technologies have accelerated the development of wireless sensor networks. The DARPA's Sensor Information Technology (SensIT) program focused on the development of a new class of software for networks of distributed microsensors. The program pursued two key thrusts: (a) development of novel networking techniques for rapidly deployable ad hoc microsensors in the battlefield, and (b) leveraging the distributed computing resources to extract right and timely information from a sensor field, including detection, classification, and tracking of targets [9].

In 1993, the University of California, Los Angeles initiated development of Wireless Integrated Network Sensors (WINS). Combining sensing, signal processing, decision making, and wireless networking capabilities in a compact, low power system, WINS was developed for monitoring and control capabilities of transportation, manufacturing, health care, environmental, and safety and security systems [11]. The advances in integrated circuit technology at the same time enabled mass production of powerful compact sensors, radios, and processors at low cost. One of the achievements of WINS is the demonstration of the feasibility of algorithms for operation of multihop wireless sensor nodes and networks at micropower level [12, 13].

In the late 1990s, the Smart Dust research project at Berkeley pursued fabrication of sensor nodes incorporating a power supply and sensing, communication, and computing hardware in a volume less than a few cubic millimeters [14, 15]. Thanks to ongoing breakthroughs in nanofabrication techniques, these sensor nodes (or motes) are expected to be the size of a grain of sand in the near future. After completion of the Smart Dust research project in 2001, as shown in Fig. 14.1, the project led to multiple follow-on research projects such as Network Embedded Systems Technology (NEST) [16], Wireless Embedded Systems (WEBS) [17], and Center for Embedded Networked Sensing (CENS) [18].

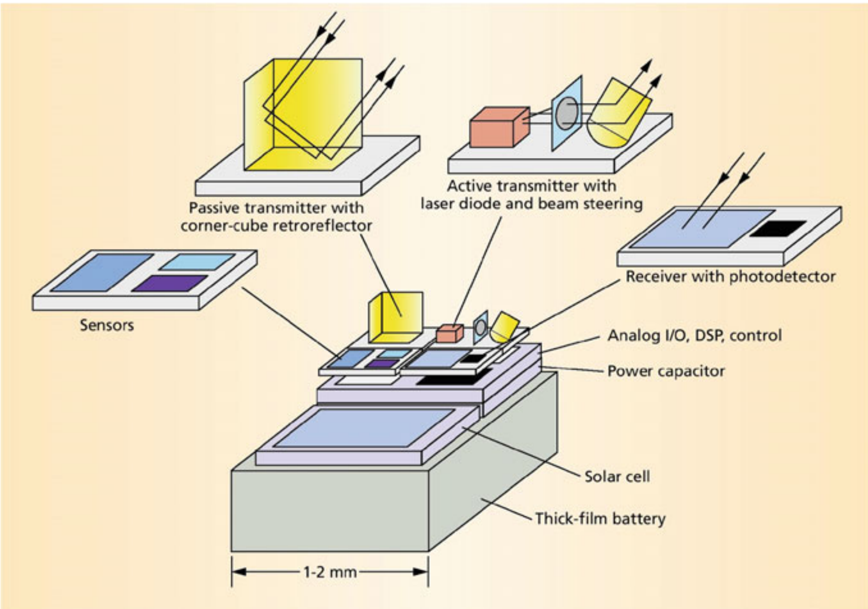

Fig. 14.1 Conceptual diagram showing a Smart Dust mote's major components: a power system, sensors, an optical transceiver, and an integrated circuit (Courtesy of [10])

14.1.2 Benefits and Challenges of Distributed Sensor Networks

In the family of DSNs, mobile DSNs have been widely used in applications for environment monitoring [19], target tracking [20, 21], and search and rescue missions [22]. Mobile sensor nodes offer advantages over static sensor networks since mobile agents can control the coverage and connectivity of the network. A typical mobile DSN consists of sensor nodes that can autonomously relocate and continuously sense, compute, and communicate. Nodes in a mobile DSN are typically scattered in space to collect information using dynamic topologies. Due to the limited communication range of sensors, however, collected information can be communicated only when each two nodes are within a communication range. Another characteristic of mobile DSNs is data distribution. In a static DSN, data can be distributed using fixed routing or flooding, while dynamic routing is used in a mobile DSN [23].

Mobile DSNs offer capabilities of distributed wireless remote sensing and processing, which translate to an improved survivability and adaptation in any environment. It is extremely difficult, for example, to conduct precise manual deployment of sensor networks for damage assessment in disaster areas or for intelligence, surveillance and reconnaissance (ISR) missions over a remote, dangerous battlefield. Nevertheless, mobile sensor nodes can proceed to areas of interest after initial deployment to complete required missions. In a surveillance and tracking mission, mobile autonomous sensor nodes can collaborate and make decisions, distributed or central, based on the shared information. For instance, as shown in Fig. 14.2, when an unmanned aerial vehicle (UAV) and an unmanned ground vehicle (UGV) are cooperatively tracking a mobile ground target, obstruction by walls for the UGV to perceive an intruder, can be overcome by the sensing performed by the UAV. The intruder location can be sent to the UGV to perform future actions. The remote independent

Fig. 14.2 The concept of cooperative target tracking using an unmanned aerial vehicle and an unmanned ground vehicle

processing capability of DSNs also increases the mission robustness. The advance of cooperative data fusion techniques, such as consensus and auction algorithms, enables a DSN to quickly combine information collected by distributed sensors.

Distributed mobile sensor networks, with their advantages, come with challenges and constraints. Challenges include deployment of resources, localization of platforms, self-organization for missions, navigation and control, allocation of tasks, energy consumption, maintenance of capabilities, and distributed data processes.

14.1.3 Cooperative Search and Tracking for Mobile Targets

Searching of mobile targets in an area using a mobile sensor network with unknown a priori target information is a challenging task. A typical formulation of confidence indicating target locations within a search area requires a grid map and an assignment of each cell in the grid with a probability value between 0 and 1. The resulting map is referred to as a probability map, which has been widely used in the past for target search and tracking. In this paper, we present a technique for cooperative search and tracking of mobile targets by a team of UAVs using probability maps. In particular, a novel strategy for updating a probability map by each agent of the sensor network is presented and validated using MATLAB simulations.

In the past, probability maps have been adopted by researchers working in the area of target search and tracking. Bertuccelli and How [24] presented a statistical framework to calculate a minimum observation time required for an agent to achieve a desired confidence level for the existence of a stationary target. They extended their work in [25] by working with slow moving targets (top speed 2 m/s). A binary value (0 or 1) was assigned to each cell based on detection results. Since agents of the network must work with independent maps, a consistent map among agents was required to store the information. Bourgault et al. [26] proposed a decentralized search strategy for a team of sensor platforms to locate a lost target based on the Bayesian rule. An optimal path planning algorithm was presented to maximize the cumulative probability of target detection. The work was extended for multiple target search [22] and tracking missions [27]. Detailed vehicle, process and observation models were adopted in [28] to validate the proposed strategy. In these scenarios, the prediction and update of a probability map were performed by a recursive Bayesian filter. However, a priori information of a lost target, such as the target's top speed and a last reported location, was assumed known. Millet et al. [29] developed a decentralized search algorithm for stationary targets. Each agent updates its individual probability map based on its observation using the Bayesian rule and performs the map fusion when other neighbors enter a space within its communication range. Mirzaei et al. [30] proposed a decentralized cooperative search and coverage algorithm for stationary targets, in which a probability map was updated using a Bayesian filter. To solve the coverage problem, the entire search region was partitioned into a Vironoi diagram and a dynamic programming method was used to obtain optimal paths for mission vehicles. Chung et al. [31] presented a framework for search and identification of multiple heterogeneous moving targets. The detection and identification formulation and update are conducted separately using the Bayesian rule. A centralized map was required to perform the optimization of the path planning and target identification. The study exposed the scalability issue for a large search area. Hu et al. [32] developed a decentralized search algorithm for stationary targets. A nonlinear transformation of a probability map was performed to reduce the communication bandwidth and a consensus-based fusion algorithm was proposed. A coverage control strategy, similar to the one used in [30], was adopted for the path planning. Finally, the asynchronous issue of the data fusion was studied.

Among the previous work for target search and tracking, the research for stationary targets has been well documented [24, 29, 30, 32], while for the ones working with moving targets, some a priori information was usually assumed known [27, 31]. A search task involving multiple moving targets using multiple sensor platforms without a priori information of targets, however, has not been well studied. The current work aims to contribute toward this unexplored area of research.

Suppose there is an unknown number of mobile ground targets to be searched and tracked by a team of unmanned aerial vehicles with gimbaled video sensors. We assume that the number of UAVs is insufficient with respect to the size of the mission area to perform a sweep search. The objective of the mission is to search for, locate, and track as many targets as possible in a mission area. To formulate the problem, we provide each UAV with a probability map of the mission area divided into cells and assign values between 0 and 1 to represent the target existence in each cell. A novel decentralized uncertainty propagation law is developed to globally update the individual probability map of each UAV by spreading the location uncertainty of mobile targets in a cell onto its neighbors using a conservative heuristic method. A measurement update step is then conducted to regionally update the probability values of cells in regions exposed by collective sensors of the UAV team. The effectiveness of the proposed strategy is validated in simulations when it was incorporated in a path planning strategy [33, 34] for search mission.

In this chapter, the shared information among UAVs is assumed to be UAV states and video sensor measurements. Sensors are assumed to have the same accuracy, and communications among UAVs are assumed to be robust, allowing UAVs to have identical probability maps throughout the search mission.

The rest of the paper is structured as follows. Section 14.2 introduces the notations used to formulate the search problem. A novel update law using a probability map in a multi-mobile-target search and tracking problem is developed in Sect. 14.3. In Sect. 14.4, algorithms for cooperative search and tracking along with the strategy for sensor placement are briefly presented. Section 14.5 presents the simulation results to validate the proposed update law for the probability map. The conclusion of our work is given in Sect. 14.6.

14.2 Problem Statement

The search area, $\mathbb{A}$, is assumed to be a plane ground and is mapped onto a set of M grid cells, $\{(x, y) \in \mathbb{A} | 0 \le x \le B_x, 0 \le y \le B_y\}$, each of which is an $\ell \times \ell$ square, where x and y are the Cartesian coordinates of the center of each cell, respectively, B_x and B_y are the boundaries of the region in the x and y directions, and ℓ is the length of the cell.

We first define the following events.

- $E_{x,y,k}$: A target is actually in cell (x, y) at time step k.
- $D_{x,y,k,i}$: A target is detected in cell (x, y) at time step k by agent i.

Each cell in $\mathbb{A}$ is associated with a probability of target existence at time step k, modeled as a Bernoulli distribution

$$\mathscr{P}\left(E_{x,y,k}\right) = q = 1 - \mathscr{P}\left(\overline{E}_{x,y,k}\right),$$

where a bar above an event denotes its complement. Then target existence can be modeled as

$$\text{Target existence} = \begin{cases} \text{A target is present} & q > 0.5, \\ \text{Unknown} & q = 0.5, \\ \text{No target is present} & q < 0.5. \end{cases} \tag{14.1}$$

Suppose that n UAVs with unique IDs, U_i, $i = 1, 2, \ldots, n$, fly above a search area with constant velocities and constant altitudes, which are assumed to be different among UAVs to avoid collisions. The position of UAV i at time step k is denoted as $\mathbf{p}^u_{i,k} \triangleq \left(p^u_{i,k,x}, p^u_{i,k,y}, p^u_{i,k,z}\right)^T \in \mathbb{R}^{3\times1}$. UAV i also maintains an individual probability map, with an initial probability value of each cell set to 0.5, indicating we are not sure of target existence. At each time step the camera sensor of agent i captures an image of the area, $\Omega_{i,k}$. UAV i is equipped with a pan-tilt camera sensor with a limited field of view (FOV), μ_i, and a limited sensing radius, R_i. The footprint of the sensor on the ground is obtained by first projecting the camera field of view (μ_i) onto the ground and then confining it by the sensing range (R_i), as shown in Fig. 14.3. At time step k, each agent i independently takes a measurement over the sensor exposure area, $\Omega_{i,k}$, denoted as the shaded region in Fig. 14.3.

The number of targets and a priori information, such as locations and movement patterns, are assumed unknown, while the target's top speed, $V^{\max}_{\text{target}}$, is assumed to be known. The target size is assumed large enough to be identified by sensors with a predefined detection probability when it enters on exposure area of the sensor.

Given the above definitions and assumptions, the problem we seek to solve is a distributed update law for the probability map on each agent based on sensor measurements. We present, in the next section, a heuristic strategy for each cell in the probability map to propagate its probability values based on the uncertainty values

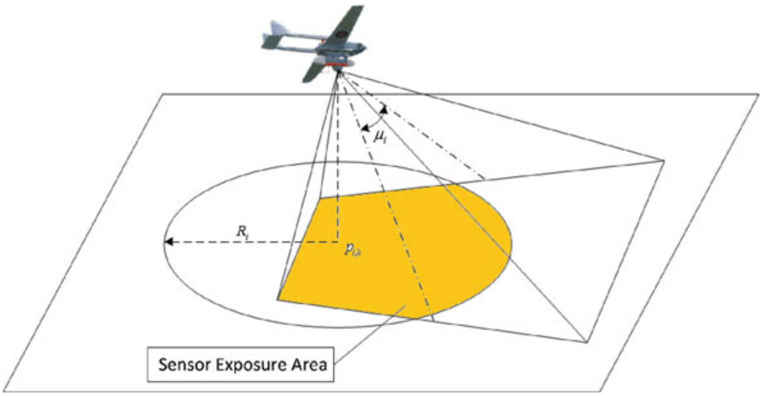

Fig. 14.3 Sensor exposure area (*shaded area*) of a UAV at $\mathbf{p}_{i,k}$ equipped with a pan-tilt video sensor with a field of view of μ_i and a limited sensing range R_i

of its neighboring cells in the previous time step. The probability values of cells in the sensor exposure area is then updated according to the measurement outcomes, such as detection or absence of targets.

14.3 Probability Map Update

For static target search, the target presence probability in a cell depends on the number of "looks" committed over the cell by a UAV team. However, when searching mobile targets in an area without a priori information, such as the number of targets and potential target locations, update rules for static search are not valid any more. In this section, we present a heuristic strategy of updating the probability map for mobile target search and tracking.

At each time step, every UAV first applies an uncertainty propagation step to update the probability value of each cell of its probability map. Depending on sensor detection results, "No target detected" or "Target detected", a measurement update step is then applied to update the probability values for cells that fall in the sensor exposure region.

14.3.1 Uncertainty Propagation Step

Since we have no a priori information, each cell with probability 0.5 may contain a target. We can designate the maximum speed of a target, $V_{\text{target}}^{\max}$, but the orientation of the target movement is unknown. Assuming that the step size of updating the probability map is T_s, the target existence probability of cell (x, y), at time k, will

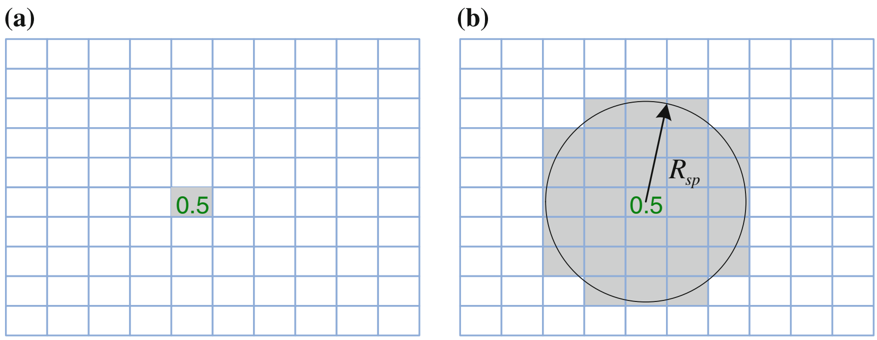

Fig. 14.4 Probability map update for a cell with the probability value of 0.5. **a** Time step k: before the update. **b** Time step $k+1$: after the update

spread, at time $k+1$, to its neighbors. This event can be described by a circle, whose center is at (x, y) and whose radius is $R_{\mathrm{sp}} = V_{\mathrm{target}}^{\max} T_s$, as shown in Fig. 14.4. The first update step of the probability in each cell is expressed as

$$\mathscr{P}_{i,k+1}(x, y) = 0.5, \quad \forall (x, y) \in \mathbb{C}, \tag{14.2}$$

$$\mathbb{C} \triangleq \left\{(x, y) \mid \left\|(x, y) - \left(x', y'\right)\right\| \leq R_{\mathrm{sp}}\right\},$$

where $\left(x', y'\right)$ is the coordinate of the center of a cell that satisfies

$$\mathscr{P}_{i,k}\left(x', y'\right) = 0.5. \tag{14.3}$$

The uncertainty propagation step can be explained as follows: at time step k, if a cell has a probability value of 0.5 (unknown target existence), the probability will spread to its neighboring cells at time $k+1$. The maximum traveling distance of a potential target equals to R_{sp}, and the area in which the potential target may stay can be expressed by a circle, defined by $\mathbb{C}$ in Eq. (14.2). Thus, at time step $k+1$, each cell in the region defined by the circle with radius R_{sp} would have an equal probability of holding that potential target. Therefore, this uncertainty should be spread out using the conservative manner in Eq. (14.2).

An issue would arise if the size of the cell (ℓ) is greater than the spreading pace (R_{sp}), which implies that one time step is not long enough to spread the probability neighboring cells. In this case, the probability value of the neighboring cells can be gradually changed to reflect the uncertainty spread. The following strategy can be used to update the probability map for such a case. The time to spread the uncertainty from a cell with probability value of 0.5 over the cells next to it can be calculated by

$$T_{\mathrm{sp}} = \frac{\ell}{R_{\mathrm{sp}}} T_s = \frac{\ell}{V_{\mathrm{target}}^{\max}},$$

where ℓ is the side length defining the cell size. The probability change of neighboring cells at each step toward 0.5 can be calculated by

$$\mathscr{P}_{\text{unit}} = \frac{0.5}{T_{\text{sp}}} = \frac{0.5V_{\text{target}}^{\max}}{\ell}. \tag{14.4}$$

Then an alternative update law is given by

$$\begin{aligned} &\mathscr{P}_{i,k+1}(x,y) = \mathscr{P}_{i,k}(x,y) + \mathscr{P}_{\text{unit}} \cdot \operatorname{sgn}\left(0.5 - \mathscr{P}_{i,k}(x,y)\right), \ \forall (x,y) \in \mathbb{C}, \\ &\mathbb{C} \triangleq \left\{(x,y) \mid \left\|(x,y) - \left(x',y'\right)\right\| \le \ell\right\}, \\ &\mathscr{P}_{i,k}\left(x',y'\right) = 0.5, \end{aligned} \tag{14.5}$$

where

$$\operatorname{sgn}(z) = \begin{cases} 1 & z > 0, \\ 0 & z = 0, \\ -1 & z < 0. \end{cases}$$

14.3.2 Measurement Update

After the uncertainty propagation step, the post process of updating the probability map will be conducted separately in two different scenarios: no detection or detection of targets. The next subsections present the detailed procedures.

14.3.2.1 No Target Detected

When no target is found at time step $k+1$ in region $\Omega_{i,k+1}$, the probability of the cells therein is updated using the following Bayesian rule [30]

$$\mathscr{P}\left(E_{x,y,k+1} | \overline{D}_{x,y,k+1,i}\right) = \frac{\mathscr{P}\left(E_{x,y,k+1}\right) \mathscr{P}\left(\overline{D}_{x,y,k+1,i} | E_{x,y,k+1}\right)}{\mathscr{P}\left(\overline{D}_{x,y,k+1,i}\right)}, (x,y) \in \Omega_{i,k+1}.$$

The probability values of the cells outside $\Omega_{i,k+1}$ remain the same. Define the probability of true positive and false positive sensor measurements as two constants, $\alpha \triangleq \mathscr{P}\left(D_{x,y,k+1,i} | E_{x,y,k+1}\right)$ and $\beta \triangleq \mathscr{P}\left(D_{x,y,k+1,i} | \overline{E}_{x,y,k+1}\right)$, respectively. The probability that a target is not detected in the cell (x,y), $\mathscr{P}\left(\overline{D}_{x,y,k+1,i}\right)$, can be calculated by [30]

$$
\begin{aligned}
\mathscr{P}\left(\overline{D}_{x,y,k+1,i}\right) &= \mathscr{P}\left(\overline{D}_{x,y,k+1,i} | E_{x,y,k+1}\right)\mathscr{P}\left(E_{x,y,k+1}\right) \\
&\quad + \mathscr{P}\left(\overline{D}_{x,y,k+1,i} | \overline{E}_{x,y,k+1}\right)\mathscr{P}\left(\overline{E}_{x,y,k+1}\right) \\
&= (1-\alpha)\,\mathscr{P}\left(E_{x,y,k+1}\right) + (1-\beta)\left(1-\mathscr{P}\left(E_{x,y,k+1}\right)\right).
\end{aligned}
$$

14.3.2.2 Target Detected

When targets are detected in region $\Omega_{i,k}$, each target will be associated with a dynamic model whose motion is predicted and updated using a Kalman filter. Target detection algorithms are assumed to be capable of distinguishing two targets that are next to each other.

Let the location and the velocity of a target at time k be $\mathbf{p}_k^t \triangleq \left(p_{k,x}^t, p_{k,y}^t\right)^T \in \mathbb{R}^{2\times 1}$ and $\mathbf{v}_k^t \triangleq \left(v_{k,x}^t, v_{k,y}^t\right)^T \in \mathbb{R}^{2\times 1}$, respectively. Selecting the system state as $\chi_k = \left(\left(\mathbf{p}_k^t\right)^T, \left(\mathbf{v}_k^t\right)^T\right)^T$ and system output at time step k as $\phi_k = \chi_k$, and assuming that the unknown system input is zero, the target dynamics is given by

$$
\begin{aligned}
\chi_{k+1} &= A\chi_k + \xi_k \\
\phi_k &= C\chi_k + \eta_k,
\end{aligned}
$$

$$
A = \begin{pmatrix} 1 & 0 & T_s & 0 \\ 0 & 1 & 0 & T_s \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix},
$$

where ξ_k is the process noise at time step k, representing modeling error and disturbances on the system, and η_k is the measurement noise at time step k, representing the sensor noise [35]. The random variables ξ_k and η_k are zero-mean Gaussian random processes with covariances Q and R, respectively.

Defining the estimated state of χ as $\hat{\chi}$ and the estimation covariance at time k as P_k, the prediction step of the Kalman filter is given by [36]

$$
\begin{aligned}
\hat{\chi}_k^- &= A\hat{\chi}_{k-1}^+ \\
P_k^- &= AP_{k-1}^+A^T + Q,
\end{aligned}
$$

where the superscripts $-$ and $+$ represent the variable values obtained before and after the update step, respectively. Defining the Kalman gain at time step k as L_k, when a measurement is available, the update step of the Kalman filter is given by

$$\begin{aligned} L_k &= P_k^- C^T \left(C P_k^- C^T + R \right)^{-1} \\ P_k^+ &= \left(I - L_k C \right) P_k^- \\ \hat{\chi}_k^+ &= \hat{\chi}_k^- + L_k \left(\phi_k - C \hat{\chi}_k^- \right), \end{aligned}$$

where I is an identity matrix.

Let the standard deviations of state estimates in the x and y directions be σ_x and σ_y, respectively, which are assumed $\sigma_x = \sigma_y$. For a Gaussian distribution, 99.73 % of the realization lies within three standard deviations of the mean [37]. Select a circular region

$$\mathbb{F} \triangleq \left\{ (x, y) \mid \left(x - \hat{p}_{k,x}^t \right)^2 + \left(y - \hat{p}_{k,y}^t \right)^2 \leq (3\sigma_x)^2 \right\},$$

whose center is at the estimated position of the target and whose radius is three times of standard deviation of the position estimation, $3\sigma_x$ (or $3\sigma_y$). Select the values for a Gaussian distribution in region $\mathbb{F}$ and rescale these values to the range [0.5, 1]. The highest value of the resulting cell in the region represents the highest confidence of target existence, while the cells with probability close to 0.5 reveal a relative low possibility of target existence. Probability values of the cells in region $\mathbb{F}$ are then updated using the scaled values accordingly.

14.4 Sensor Management and Path Planning for Search and Tracking

The objective of search is to maximize the possibility of detecting targets in the region of interest, while the objective of tracking is to minimize the overall uncertainty of target locations, i.e., the trace of the estimation covariance matrix [35] of the targets, which have been found. For UAVs equipped with pan-tilt video sensors, two tasks need to be solved: the sensor management, i.e., which orientation the sensor should point towards, and the path planning.

14.4.1 Search Mode

Until a target is found, UAVs operate in the search mode. Due to the projection distortion effect of video sensors, the gimbal is commanded to consistently point downward to obtain the best resolution of the image. For the path planning algorithm, we use the guidance law for multiple UAVs searching multiple mobile targets, reported in [33, 34]. The desired heading angle of UAV i, ψ_i, can be calculated by the following equation.

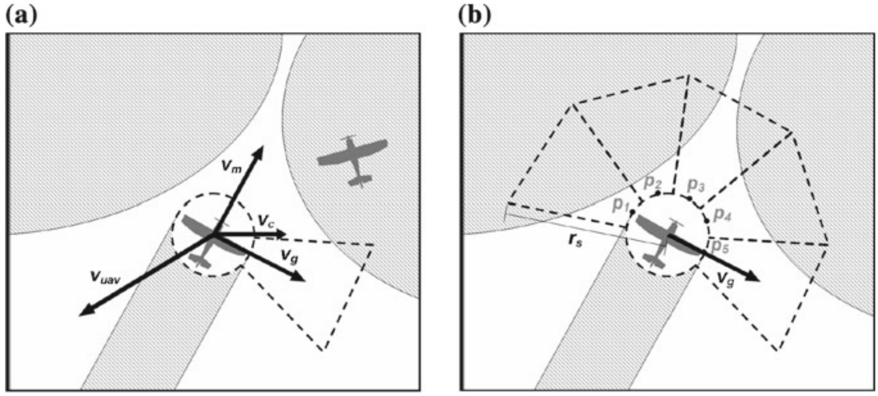

Fig. 14.5 An optimal path planning algorithm for cooperative target search. **a** The four vectors involved in the determination of the desired heading of a UAV. **b** The unitary vector v_g points in the direction of the *trapezoid-shaped* segment (one out of five in this case) with the highest probability of target detection (*white area*) [33]

$$\psi_i = \omega_g v_g + \omega_{\mathrm{UAV}} v_{\mathrm{UAV}} + \omega_c v_c + \omega_m v_m, \tag{14.6}$$

where v_g is the goal vector that points in the direction of the area around the UAV that has the largest probability of finding a target, as depicted in Fig. 14.5; v_{UAV} is a vector responsible for UAV to UAV collision avoidance; v_c is a vector used to maintain the UAV within a predetermined search area by operating as an obstacle avoidance vector that concerns itself only with search boundaries within a radial comfort range around the UAV; v_m is a momentum vector that incorporates in ψ_i to maintain the previous heading; and ω_g, ω_{UAV}, ω_c, and ω_m are corresponding weights for aforementioned vectors, respectively.

A set of optimized weights is given in [33] as $\omega_g = 0.63$, $\omega_{\mathrm{UAV}} = 0.26$, $\omega_c = 0.27$, and $\omega_m = 0.03$. It can be seen that the largest weight ω_g makes v_g play the most significant role among vectors in Eq. (14.6) of determining the desired heading angle for UAV i. Therefore, a proper dynamic probability map is essential for a UAV team to maximize the probability of detecting targets in a search area.

14.4.2 Tracking Mode

When a target is found, UAVs switch to the tracking mode. It is noted that there are other operating modes, such as validating target, re-acquiring target, and approaching target, which are beyond the scope of this work. For a UAV tracking multiple targets, sensor management plays a significant part in decreasing the uncertainty of target states. The path planning algorithm for UAVs to cooperatively track multiple mobile targets is also a challenging problem. To use the sensor management algorithm developed in [38] and the path planning strategy in [39] effectively, which are for multi-target tracking using a single UAV, the targets that have been detected are allocated into N_u groups dynamically based on their relative distances. The resulting N_u groups are then assigned to the nearest UAVs' allowing each UAV to perform the sensor placement and path planning tasks, accordingly.

14.5 Simulation Results

This section presents MATLAB simulation results of performing distributed cooperative search and tracking of ground mobile targets in an unknown environment using a team of two UAVs. As shown in Fig. 14.6, the mission area of search and tracking targets and allowed flying zone are enclosed by two large (black and red, respectively) rectangles. Two search areas, a circular region on the left and a rectangular region on the right (blue lines), are specified by a user. Two UAVs are commanded out from the home position (the origin) toward the search areas for a search and track mission. Each UAV is depicted as a small colored circle. The gray thin tail attached to each UAV represents the historical UAV trajectory. The gray rectangle around the head of each UAV represents the sensor exposure area. In Fig. 14.6, the UAVs are in the "Fly to Search Area" mode, so the gimbal is regulated pointing downward, resulting in square sensor exposure areas. In this simulation, a total of three targets, shown as small colored rectangles, are making random walk movements in the mission area. The two subfigures arranged vertically on the right side are probability maps for the two UAVs, respectively. The probability values are displayed using a "gray" colormap defined in MATLAB, which maps value 0 to color black, value 1 to color white, and value 0.5 to color gray. The title line of each probability map shows the unique UAV ID and its current mode.

Key parameters used in the simulation are presented in Table 14.1. The simulation step size T_S was selected as 0.3 s. The actual top speed of the targets in the simulation

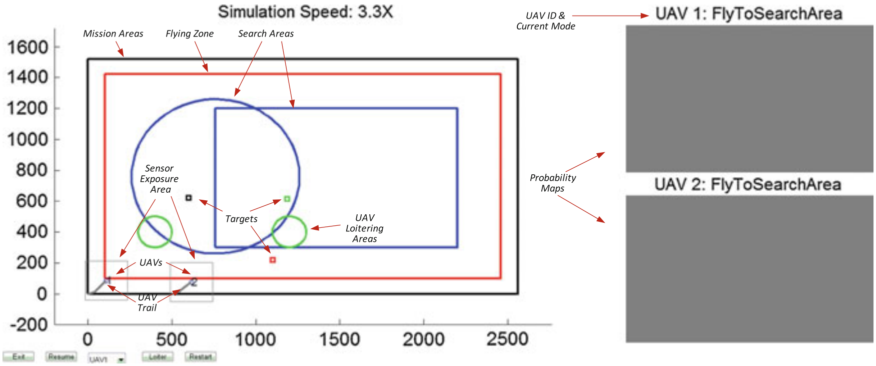

Fig. 14.6 Simulation snapshot: initialization

Table 14.1 Simulation parameters

Map size		UAVs	
Mission area (m^2)	2,560 × 1,520	Speed (m/s)	20
Flying zone (m^2)	2,360 × 1,320	Minimum turning radius (m)	100
Cell size (m^2)	20 × 20	Sensor range radius (m)	300
Search area 1 (m^2)	$\pi \cdot 500^2$	Sensor FOV (deg)	56
Search area 2 (m^2)	1,440 × 900	Sensor slew rate (deg/s)	150

was 10 m/s. R_{sp} is then calculated as 3 m, which is smaller than the side length of the cell, ℓ (20 m). Therefore, the update law (14.5) should be used. The spread pace $\mathscr{P}_{unit}$ is then calculated by using Eq. (14.4) as $\mathscr{P}_{unit} = 0.075$. The probability map is recursively updated at each time step by using the strategy proposed in Sect. 14.3. To simplify the calculation, we select $\alpha = 1$ and $\beta = 0$ in the simulation.

Figures 14.7 and 14.8 present a series of snapshots of the simulation run in which two UAVs searched and tracked three targets in an unknown area. To simplify the problem, UAVs are assumed to communicate all of their knowledge regarding targets

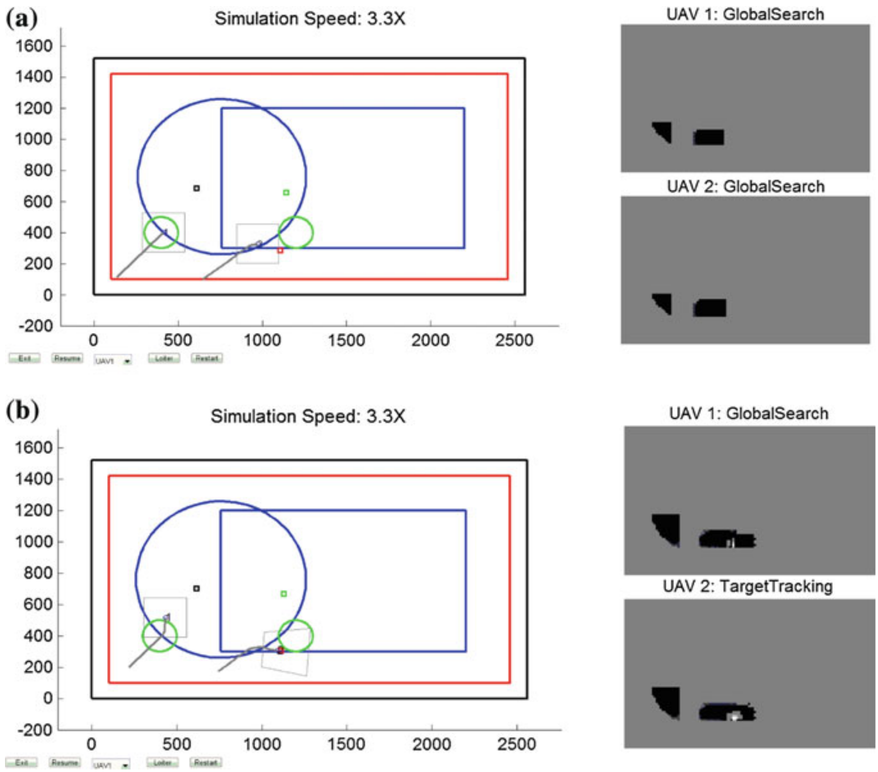

Fig. 14.7 Group one of simulation snapshots: search and tracking of three mobile targets using two UAVs. **a** UAVs enter search areas ($t = 10$ s). **b** One UAV finds a target ($t = 11$ s)

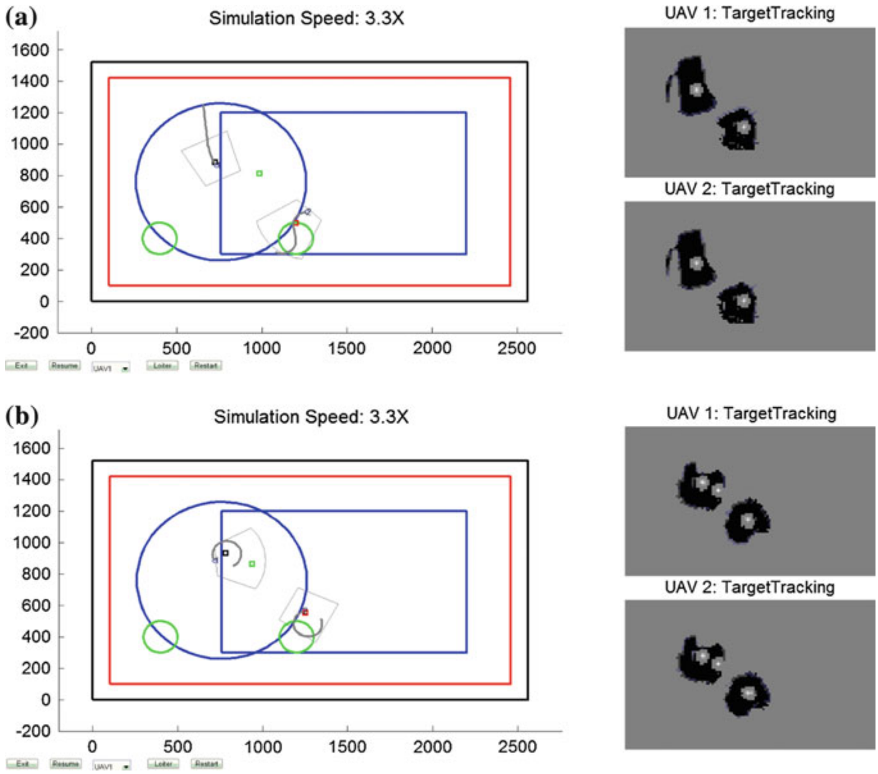

Fig. 14.8 Group two of simulation snapshots: search and tracking of three mobile targets using two UAVs. **a** Two UAVs find two targets. ($t = 30$ s). **b** Two UAVs track three targets. ($t = 66$ s)

and the UAV status. After the two UAVs take off from the base, as shown in Fig. 14.6, they fly toward their search areas. As shown in Fig. 14.7a, after UAVs reach the search area, the "Global Search" mode is initiated. It can be seen that due to the use of the path planning algorithm proposed in Sect. 14.4.1, the two UAVs move apart from each other to maximize the coverage of the search area. The probability map is updated accordingly by using the strategy described in Sect. 14.3. As shown in Fig. 14.7b, when a UAV finds a target, it switches to the "Target Tracking" mode, while the other UAV is still operating in the "Global Search" mode. When the two UAVs both find targets, as shown in Fig. 14.8a, the cooperative tracking algorithm proposed in Sect. 14.4.2 is applied accordingly. Figures 14.8b shows that the two UAVs are able to track three targets. The probability maps of the two UAVs are updated accordingly using the strategy proposed in Sect. 14.3.

To further demonstrate the effectiveness of the proposed target search strategy, we conducted a series of simulations where three targets are randomly placed in the search area and are commanded to move along straight paths with constant velocities. Table 14.2 lists ten data sets of target initial locations and velocities and time periods that the UAV team took to locate the targets. However, when a UAV detects a target,

Table 14.2 Simulation configuration for two UAVs to perform search and track missions

Target initial locations (m)	Target velocities (m/s)	Time (s)
(2300, 1300), (2300, 1300), (2300, 1300)	(−2, 0), (−4, −2), (0, −5)	106
(600, 600), (1100, 200), (1200, 600)	(0, 4), (−0.5, 5), (−3, −3)	29
(200, 200), (800, 200), (600, 300)	(2, 1), (0.5, 3), (1, 5)	135
(1500, 800), (1500, 800), (1500, 800)	(−2, 1), (3, 3), (5, −5)	128
(100, 100), (200, 1300), (2300, 200)	(5, 4), (5, −2), (−4, 2)	21
(350, 100), (200, 900), (1000, 500)	(5, 0), (4, −1), (3, 2)	90
(500, 800), (500, 800), (500, 800)	(5, 0), (4, −1), (3, 2)	83
(1300, 1300), (1300, 1300), (1300, 1300)	(0, −5), (2, −5), (−3, −4)	77
(2300, 800), (2300, 800), (2300, 800)	(−5, 0), (−4, −1), (−3, 1)	100
(1300, 1300), (2300, 800), (200, 800)	(0, −4), (−6, 0), (5, 0)	47

it switches to the tracking mode in which the UAV keeps tracking detected target(s) using the strategy introduced in Sect. 14.4.2. So the "Time" column in Table 14.2 lists the time periods starting from the beginning of the simulation until all three targets are found or until two UAVs both switch to the tracking modes. It can be seen from Table 14.2 that both UAVs are able to detect targets in all cases and the maximal time used is 135 s in a mission area with dimension of 2,5604 $\times$ 1,520 m^2.

14.6 Conclusion

This paper presents the use of mobile sensor network to search and track multiple mobile targets in an unknown area. The network is made of a team of UAVs equipped with pan-tilt video sensors. We introduced a novel update strategy for the probability map used by UAVs to store probability information of target locations in the search area. Two update laws are proposed to accommodate maps with different scales. The simulation results show the rationality of the proposed probability-map update strategy. The future work will focus on the comparison study and cooperative tracking guidance law for multiple UAVs.

References

1. Dargie W, Poellabauer C (2011) Fundamentals of wireless sensor networks: theory and practice. Wiley, New York
2. Estrin D, Girod L, Pottie G, Srivastava M (2001) Instrumenting the world with wireless sensor networks. In IEEE international conference on acoustics, speech, and signal processing (ICASSP), vol 4, pp 2033–2036
3. Sohraby K, Minoli D, Znati T (2007) Wireless sensor networks: technology, protocols, and applications. Wiley, Hoboken
4. Bryan C, Nishimura C (1995) Monitoring oceanic earthquakes with SOSUS: an example from the caribbean. Oceanography 8(1):4–10

5. Nishimura C, Conlon D (1994) IUSS dual use: monitoring whales and earthquakes using SOSUS. Mar Technol Soc J 27(4):13–21
6. Chong C, Kumar S (2003) Sensor networks: evolution, opportunities, and challenges. Proc IEEE 91(8):1247–1256
7. Rashid R, Julin D, Orr D, Sanzi R, Baron R, Forin A, Golub D, Jones M (1989) Mach: a system software kernel. In: IEEE computer society international conference: intellectual leverage, digest of papers, Feb 1989, pp 176–178
8. Rashid RF, Robertson GG (1981) Accent: a communication oriented network operating system kernel, vol 15, New York, pp 64–75
9. Kumar S, Shepherd D (1993) SensIT: sensor information technology for the warfighter. In: Proceedings of 4th international conference on information fusion, pp TuC1-2–TuC1-9
10. Warneke B, Last M, Liebowitz B, Pister KSJ (2001) Smart Dust: communicating with a cubic-millimeter computer. Computer 34(1):44–51
11. Asada G, Dong M, Lin TS, Newberg F, Pottie G, Kaiser WJ, Marcy HO (1998) Wireless integrated network sensors: low power systems on a chip. In: Proceedings of the 24th European solid-state circuits conference, Sept 1998, pp 9–16
12. Agre JR, Clare LP, Pottie GJ, Romanov NP (1999) Development platform for self-organizing wireless sensor networks. In: Aerosense 1999, Orlando, April 1999
13. Pottie GJ, Kaiser WJ (2000) Wireless integrated network sensors. Commun ACM 43(5):51–58
14. Kahn JM, Katz RH, Pister KSJ (1999) Next century challenges: mobile networking for smart dust. In: International conference on mobile computing and networking, pp 271–278
15. Kahn JM, Katz RH, Pister KSJ (2000) Emerging challenges: mobile networking for smart dust. J Commun Netw 2(3):188–196
16. http://webs.cs.berkeley.edu/nest-index.html. Accessed on 20-May-2014
17. http://smote.cs.berkeley.edu:8000/tracenv/wiki. Accessed on 20-May-2014
18. http://research.cens.ucla.edu/research/. Accessed on 20-May-2014
19. Corke P, Wark T, Jurdak R, Hu W, Valencia P, Moore D (2010) Environmental wireless sensor networks. Proc IEEE 98(11):1903–1917
20. Atia GK, Veeravalli VV, Fuemmeler JA (2011) Sensor scheduling for energy-efficient target tracking in sensor networks. IEEE Trans Signal Process 59(10):4923–4937
21. Hongju C, Su G, Jiannan H (2012) A distributed cross-layer framework for target tracking in three-dimensional wireless sensor networks. Future wireless networks and information systems, Lecture Notes in Electrical Engineering, vol 143, Springer, Berlin, pp 653–661
22. Wong EM, Bourgault F, Furukawa T (2005) Multi-vehicle bayesian search for multiple lost targets. In: IEEE international conference on robotics and automation, Apr 2005, pp 3169–3174
23. Yick J, Mukherjee B, Ghosal D (2008) Wireless sensor network survey. Comput Netw 52(12):2292–2330
24. Bertuccelli LF, How JP (2005) Robust UAV search for environments with imprecise probability maps. In: 44th IEEE conference on decision and control and 2005 European control conference, Dec 2005, pp 5680–5685
25. Bertuccelli LF, How JP (2006) Search for dynamic targets with uncertain probability maps. In: American control conference, June 2006
26. Bourgault F, Furukawa T, Durrant-Whyte HF (2003) Coordinated decentralized search for a lost target in a Bayesian world. In: IEEE/RSJ international conference on intelligent robots and systems, vol 1, Oct 2003, pp 48–53
27. Furukawa T, Bourgault F, Lavis B, Durrant-Whyte HF (2006) Recursive Bayesian search-and-tracking using coordinated UAVs for lost targets. In: IEEE international conference on robotics and automation, May 2006, pp 2521–2526
28. Bourgault F, Furukawa T, Durrant-Whyte HF (2006) Optimal search for a lost target in a Bayesian world. Field and service robotics, Springer tracts in advanced robotics, vol 24. Springer, Berlin, pp 209–222
29. Millet PT, Casbeer DW, Mercker T, Bishop JL (2010) Multi-agent decentralized search of a probability map with communication constraints. In: AIAA guidance, navigation and control conference

30. Mirzaei M, Sharifi F, Gordon BW, Rabbath CA, Zhang YM (2011) Cooperative multi-vehicle search and coverage problem in uncertain environments. In: IEEE conference on decision and control and European control conference, Dec 2011, pp 4140–4145
31. Chung TH, Kress M, Royset JO (2009) Probabilistic search optimization and mission assignment for heterogeneous autonomous agents. In: IEEE international conference on robotics and automation, May 2009, pp 939–945
32. Hu J, Lihua X, Kai-Yew L, Xu J (2013) Multiagent information fusion and cooperative control in target search. IEEE Trans Control Syst Technol 21(4):1223–1235
33. DeLima P, Pack D (2008) Toward developing an optimal cooperative search algorithm for multiple unmanned aerial vehicles. In: International symposium on collaborative technologies and systems, May 2008, pp 506–512
34. DeLima P, Pack D, Jr Sciortino JC (2007) Optimizing a search strategy for multiple mobile agents. In: Society of photo-optical instrumentation engineers (SPIE) conference series, vol 6563
35. Beard RW, McLain TW (2012) Small unmanned aircraft: theory and practice. Princeton University Press, Princeton
36. Maybeck PS (1979) Stochastic models, estimation, and control, Mathematics in Science and Engineering. Academic press, Boston
37. Pukelsheim Friedrich (1994) The three sigma rule. Am Stat 48(2):88–91
38. Farmani N, Sun L, Pack D (2014) An optimal sensor management technique for unmanned aerial vehicles tracking multiple mobile ground targets. In: International conference on unmanned aircraft systems, Orlando, May 2014
39. Farmani N, Sun L, Pack D (2014) Optimal UAV sensor management and path planning for tracking multiple mobile targets. In: ASME 2014 dynamic system and control conferences, San Antonio, October 2014