This is the author's version of an article that has been published in IEEE Transactions on Cybernetics.
DOI: 10.1109/TCYB.2018.2875625
The final version of the paper is available at https://ieeexplore.ieee.org/document/8509163

A Hybrid Strategy for Target Search Using Static and Mobile Sensors

Zendai Kashino, *Student Member*, *IEEE*, Goldie Nejat, *Member*, *IEEE*, and Beno Benhabib

***Abstract*—Locating a mobile target, untrackable in real-time, is pertinent to numerous time-critical applications, such as wilderness search and rescue. This paper proposes a hybrid approach to this dynamic problem, where both static and mobile sensors are utilized for the goal of detecting the target. The approach is novel in that the team of robots utilized to deploy a static-sensor network also actively searches for the target, via on-board sensors. Synergy is achieved through (*i*) the optimal deployment planning of the static-sensor network, as well as (*ii*) the optimal routing and motion planning of the robots for the deployment of the network and target-search.**

The static-sensor network is planned first to maximize the likelihood of target detection, while ensuring (temporal and spatial) unbiasedness in target motion. Robot motions are, subsequently, planned in two stages: route planning, and trajectory planning. In the first stage, given a static-sensor network configuration, robot routes are planned to maximize the amount of spare time available to the mobile agents/sensors, for target search in between (*just-in-time*) static-sensor deployments. In the second stage, given robot routes (i.e., optimal sequences of sensor delivery locations and times), corresponding robot trajectories are planned to make effective use of any spare time the mobile agents may have to search for the target.

The proposed search strategy was validated through extensive simulations, some of which are detailed herein. Included is also an analysis of the method's performance in terms of target-search success.

***Index Terms*—Hybrid search planning, mobile-target search, multirobot coordination, wilderness search and rescue (WiSAR), wireless sensor networks.**

I. Introduction

THE mobile-target search problem is pertinent to numerous real-world situations, including various forms of search and rescue, where the target is, typically, un-trackable (i.e., his/her location is unknown in real-time) [1]-[8]. Planning a search for such a target would require coordinating the available search resources to maximize the likelihood of detection [9]-[12].

A search for a mobile target is, generally, performed by agents who actively move through the search space to locate it [12]-[26]. For example, in [22], a method is presented for coordinating robot formations sweeping the entirety of a search area populated with arbitrarily-shaped obstacles. In [23], a technique for dynamically reconfiguring the search space is discussed. The technique enables autonomous search, as well as the tracking of targets that may move outside of the initially defined search boundaries. In [24], a search technique guided by recursive Bayesian estimation of the target's location is proposed.

Mobile robots, while representing a reconfigurable network, are expensive to deploy and operate as search resources. Static sensors, on the other hand, allow the monitoring of a large geographical area at a significantly lower cost.

There have been research papers that recommend the use of only static search resources, for example, in the context of surveillance, [27]-[33]. Though, most do not explicitly deal with dynamic scenarios, where the target may not be guaranteed to pass through the region of interest, nor do they consider expanding the search area. As a modified approach, a time-phased deployment of the static-sensor network was proposed in [34], [35]. It allows for the network deployment plan to be changed in mid-search, adapting to new information regarding the target, were it to become available.

The combined use of mobile and static resources has also been investigated, though, for applications with mainly static elements (e.g., fixed regions of interest, pre-configured static-sensor networks, etc.) [36]-[42]. For example, in [37], mobile agents are used to patrol regions not covered by the static-sensor network. In [38]-[40], mobile agents are simply used to service the static-sensor network. In [41], mobile agents are used to deploy and/or acquire data from static sensors, eliminating the need for a wirelessly connected network.

The aforementioned methods, typically, do not consider dynamic scenarios wherein the region of interest may change over time, and only utilize one of the two resources for search. In time-critical applications, however, it is essential to maximize resource utilization and efficiency to increase the likelihood of the target being located as soon as possible.

This paper, thus, presents a novel mobile-target search method that uses a dynamically deployed static-sensor network supported by a robot team. The strategy is unique in that the robot team deploying the sensor-network also actively searches for the target between sensor deliveries.

The proposed method was developed with, primarily, application to real-time wilderness search and rescue (WiSAR) planning in mind. Namely, it considers the problem

This research was funded in part by the Natural Sciences and Engineering Research Council of Canada (NSERC), and the Canada Research Chairs program.

All authors are with the Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, ON M5S 3G8, Canada. (zendkash@mie.utoronto.ca)

DOI: 10.1109/TCYB.2018.2875625
The final version of the paper is available at https://ieeexplore.ieee.org/document/8509163

in which the objective is to locate a lost person, wandering in the wilderness, as soon as possible [19], [43]-[45]. The generalized method, however, can be applied to any mobile-target search problem in which knowledge regarding the target's motion can change in real-time. Some examples of other similar search problems include surveillance [46], wildlife search [47], target pursuit [48], [49], and urban search and rescue [50]-[54]. In target pursuit, for example, the aim is to locate an evading target. Similarly, in urban search and rescue, the aim is to locate a disaster survivor who may be wandering in the aftermath. In each case, search is time-critical and dynamic such that it could benefit from maximum resource use and adaptability.

The proposed method first plans an optimal static-sensor network, maximizing its flexibility and likelihood of locating the target. Thereafter, it plans the motion of the mobile agents (i.e., robots) to deliver sensors and to also actively search for the target.

II. Mobile-Target Search Problem

This paper addresses the mobile-target search problem, within a region that grows with time, using a dynamically deployed static-sensor network supported by an autonomous robot team (e.g., WiSAR). The latter performs both target search and deployment of the static-sensor network to maximize resource utilization. Section II.A below presents the overall system model and assumptions made. It is followed by the problem formulation in Section II.B.

A. System Model and Assumptions

In WiSAR-type problems, the objective is to locate a lost person in a search area expanding with time. Planning a search would, thus, require effective coordination of search resources (mobile robots and static sensors in our case) to locate the target as soon as possible based on the available information prior to and during the search [55], [56]. Herein, the search is modelled to occur in 2D continuous space. The target is modelled as a point of interest and follows an unknown trajectory through the search area. Namely, its exact location is unknown until it is detected. When the target is detected, the search terminates. Robots and static-sensors are also modelled as points for the sake of simplicity. Furthermore, robot and sensor coverage of the search area is modelled with a Boolean disc coverage model. Namely, a target is detected if it passes within a given distance of a static sensor or robot. It should be noted that, while the search is modelled to be in a 2D space, 3D constructs such as terrain (elevation, vegetation, etc.) are considered when planning search.

The assumptions made in our work include:

Assumption 1: Static sensors are deployed by robots. Namely, sensors cannot move on their own. Furthermore, they are non-retrievable, once deployed, and can thus not be relocated.

Assumption 2: All robots/sensors are assumed to have global access to a central controller coordinating the hybrid search. Namely, the proposed method is a centralized one in which robots and static sensors do not communicate with each other directly. Instead, they all communicate with a central controller that coordinates their movements and receives reports regarding search observations. Furthermore, the communication is assumed to be flawless such that all units are always connected, and all messages are transmitted perfectly.

Assumption 3: Search scenario information known at the start of the search includes the target demographics, its motion characteristics and last known position (LKP), as well as search-area terrain, and search-resource availability. This information is assumed to either be publicly available or provided by individuals requesting the search. Additional information about the target can be discovered at any time during the search. For example, a clue left by the target (e.g., an article of clothing) could serve as a new LKP. New information can change the optimal search plan.

Assumption 4: A mobility model that can generate realistic target motion is available. Herein, it is assumed that the target propagates outward from the LKP while 'wandering'. Target motion is specified by two parameters, d_{max} and σ_θ. d_{max}, specifies the maximum distance the target can move before changing directions; while σ_θ specifies the degree to which the target wanders as it propagates away from the LKP. Values for d_{max} and σ_θ can be inferred from search-scenario information. This target model was adopted from [35] and is based on lost-person behaviour in the wilderness. Other motion models could also be used (e.g., [2], [35], [43], [57], [58]).

Assumption 5: A model for determining the effect of terrain (e.g., slopes, vegetation, etc.) on the target's and robots' speeds is available.

Assumption 6: The search area increases in size with time, as the target propagates outwards. However, the target always remains within the sensing field (i.e., search area) throughout the search. Namely, the proposed method plans robot search and static-sensor deployment at increasingly larger radii away from the LKP over time to account for the growing of the search area.

B. Problem Formulation

The overall problem of planning a hybrid search can be divided into three interconnected sub-problems (phases): (*i*) sensor-network deployment planning, (*ii*) sensor-delivery route planning, and (*iii*) robot-trajectory planning. These are individually detailed and formulated below.

1) Static-Sensor-Network Deployment Planning

In this phase, the goal is to determine an optimal network configuration for n sensors, defined by their deployment locations, $\{(x_1, y_1), \cdots, (x_n, y_n)\}$, and corresponding deployment times, $\{t_1, \cdots, t_n\}$. The latter refer to times at which sensors should be delivered to their corresponding optimal locations by a robot.

In the optimal network configuration, the search effort needs to be spread throughout the search unbiasedly, while maximizing the likelihood of target detection [35]. Namely, the network must account for all possible target-motion directions since this would be *a priori* unknown.

DOI: 10.1109/TCYB.2018.2875625
The final version of the paper is available at https://ieeexplore.ieee.org/document/8509163

Network-deployment planning can be carried out in two stages. In the first stage, sensor-deployment times need to be determined such that the rate of deployment does not favor any specific period of time. In the second stage, sensor-deployment locations need to be determined to maximize time-cumulative likelihood of detection in a spatiotemporally unbiased manner.

Stage 1: Determining optimal sensor-deployment times

As static sensors cannot be relocated after deployment, a fully-deployed network would not be reconfigurable to adapt to any new information that may become available during the search. Thus, deploying sensors in a time-phased manner during the search would allow the (undeployed part of the) network to be reconfigured using the sensors not yet deployed.

In order to maximize the network's ability to adapt to target information obtained during the search, deployment times must be optimized such that the network is temporally unbiased. Temporally-unbiased sensor-deployment times are ones that spread search effort uniformly. This can be achieved by determining placement times that approximate a uniform rate of search-effort deployment, up to time t: i.e., a linear cumulative effort function, $E(t)$.

The objective of deployment-time optimization, then, can be formulated as a minimization of error between the empirical $E(t)$ and its ideal form:

$$\text{minimize } f_{Dep\_time} = \int_{t_1}^{t_n} |E(t) - (At + B)|\, dt, \qquad (1)$$

where A and B are constants.

Deployment times, $\{t_1, \cdots, t_n\}$, are varied and optimized to minimize Eq. (1). The optimization requires the first and last deployment times, t_1 and t_n, to be known as they need to be fixed. Otherwise, this problem would be an unconstrained optimization problem.

Stage 2: Determining optimal sensor-deployment locations

Once deployment times have been determined, the corresponding sensor-deployment locations that maximize the likelihood of target detection in a directionally unbiased manner need to be determined. This problem can be formulated as two independent optimizations: (1) determining unique radial locations for sensors that maximize the likelihood of target detection over a sensor's lifetime, and (2) determining unique angular locations for sensors that best approximate an ideal spatiotemporal sensor distribution for spatiotemporal unbiasedness.

In order to formulate an expression for the likelihood of target detection over a sensor's lifetime, i.e., *the time-cumulative likelihood of target detection*, let us assume there exists a probability density function describing where the target may be in the search area at any given time, t, $\rho(x, y, t)$. The first objective function is, then, defined by:

$$\text{maximize } f_{Dep\_rad} = \int_{t_i}^{t_e} \rho(x, y, t)\, dt, \qquad (2)$$

where t_i is the deployment time of the i^{th} sensor and t_e is the end of search time.

The second objective function is defined by:

$$\text{minimize } f_{Dep\_ang} = \int_0^{2\pi} \int_{t_1}^{t_n} \left|F(\theta, t) - \hat{F}(\theta, t)\right|^2 dt d\theta, \qquad (3)$$

where F and $\hat{F}$ are the ideal and empirical spatiotemporal distributions of sensor deployment locations and times..

The combined optimization is, thus, one in which radial sensor deployment locations are optimized to maximize Eq. (2), f_{Dep_rad}, and angular deployment locations are optimized to minimize Eq. (3), f_{Dep_ang}. The outcome would yield a set of optimal deployment locations, $\{(x_1, y_1), \cdots, (x_n, y_n)\}$, maximizing the time-cumulative likelihood of target detection in a spatiotemporally unbiased manner. Optimizing sensor deployment locations for spatiotemporal unbiasedness has the side effect of spatially spreading sensors such that redundant coverage of the search area is minimized. Both optimizations are unconstrained as sensors may be deployed anywhere in the search area as long as their coverage areas do not overlap.

2) Robot-Route Planning

In *robot-route planning*, the goal is to determine optimal routes for the k search robots that would deploy the optimal static-sensor network. The problem involves task-allocation (i.e., deciding which robots will deploy which sensors) as well as determining the individual robot routes (i.e., strings/sequences of sensor nodes to be visited) [59], [60].

For the optimization, the primary objective function is the maximization of spare times between sensor deployments. *Spare time* refers to how early a robot can arrive at a sensor deployment location. The more spare time a robot would have, the more time it could spend searching for the target, while still arriving at the next delivery node *just-in-time*.

Let us assume a robot is tasked to deploy Sensor j, following its previous deployment of Sensor i, at times t_j and t_i, respectively. Also, let the earliest time a robot can arrive at the deployment location (x_j, y_j) be denoted by t_{aj}, travelling at the fastest speed possible for a duration of Δt_{ji}. Then, *spare time* can be calculated as:

$$\delta t_j = t_j - t_{aj}, \qquad (4)$$

where

$$t_{aj} = t_i + \Delta t_{ij}. \qquad (5)$$

The objective function for route planning can, then, be defined as the maximization of the minimum spare time over all sensors:

$$\text{maximize } f_{Route} = \min_{j \in \{1, \cdots, n\}} \delta t_j, \qquad (6)$$

and/or, alternatively, as the maximization of the total spare time over all sensors:

$$\text{maximize } f_{Route\_alt} = \sum_{j \in \{1, \cdots, n\}} \delta t_j. \qquad (7)$$

The above optimization, thus, is one in which a set of deployment sequences of sensor nodes, one per each robot, $S = \{S_1, \cdots, S_k\}$, is chosen to maximize spare time. Here, S_r denotes the sequence of sensors to be visited by Robot r (e.g., $S_1 = \{1, 3, 8\}$ denotes that Robot 1 will deliver Sensors 1, 3, and 8, in this given order). There are no constraints on this optimization in the context of this work.

3) Robot-Trajectory Planning

In *robot-motion trajectory planning*, after optimal robot routes have been obtained, the goal is to determine the respective trajectories for optimal target search. Namely, once a robot has been assigned to deploy Sensor j, after deploying Sensor i, its corresponding motion is planned such that it optimally searches for the target between these deployments.

The optimal search trajectory, planned herein, is one that has the robot searching for the target while remaining on its respective *iso-probability curve* that is propagating forward with time [21], [61]. Iso-probability curves are constructed assuming a probability distribution describing *probable* target propagation speeds away from the LKP. In brief, the P% iso-probability curve delimits the extent to which the slowest P^{th} percentile target could travel away from the LKP. This boundary propagates with time, as targets would have more time to travel further away from the LKP.

A set of iso-probability curves would be selected for robots to remain on during the search (e.g., Robot 1 staying on the 10% curve, Robot 2 staying on the 30% curve, etc.). Fig. 1(a) and 1(b) show examples of (propagating) 30%, 50%, and 70% iso-probability curves at times t and $t+\Delta t$, respectively.

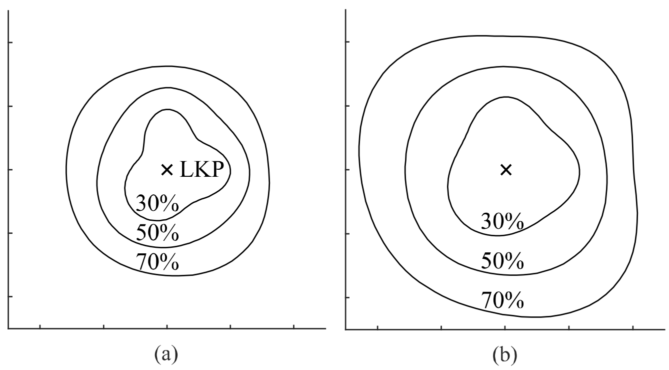

Fig. 1. Three example iso-probability curves at *(a)* Time t, and *(b)* Time $t+\Delta t$.

Searching for the target while remaining on these curves allows for the distribution of search effort to increase the likelihood of target detection. One can note that, remaining/travelling on the propagating curves that encircle the LKP allows robots to look for the target in all possible directions of target motion. This is desired since, herein, there are no assumptions made on the target travel direction.

However, robots cannot always remain on their respective iso-probability curve due to their need to deploy sensors at specified times, which cannot be synced with the passing of the curves through the sensors nodes' optimal locations. Thus, a robot trajectory between deployments needs to be divided into three segments: (*i*) a segment leading from Sensor-node i to the (earliest possible) starting point on the iso-probability curve, (*ii*) a segment defining the (optimal) search trajectory (i.e., the trajectory followed by the robot while remaining on its respective iso-probability curve), and (*iii*) a segment connecting the (latest possible) departure point from the iso-probability curve moving toward Sensor-node j, Fig. 2. Segment *ii* in Fig. 2, represents a time-phased collection of a robot's positions while remaining on its respective iso-probability curve that is propagating with time.

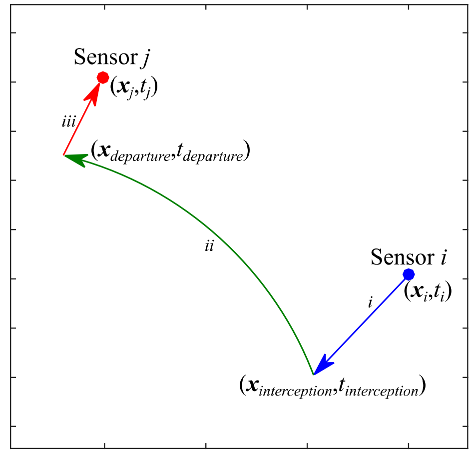

Fig. 2. An example overall robot trajectory between two sensor nodes.

Since the optimal search trajectory begins when the robot *intercepts* its assigned iso-probability curve and ends when it *departs* from this curve, the three segments connect four waypoints: the Sensor i location, $\boldsymbol{x}_i$, the iso-probability curve interception location, $\boldsymbol{x}_{interception}$, the iso-probability curve departure location, $\boldsymbol{x}_{departure}$, and the Sensor j location, $\boldsymbol{x}_j$. Two of the four waypoints are known. Thus, the problem consists of optimizing the intermediate waypoints to maximize the time spent on the optimal search trajectory.

Let $t_{interception}$ be the time at which the robot starts the optimal *search segment* at $\boldsymbol{x}_{interception}$ (i.e., the time at which the robot intercepts the iso-probability curve), and $t_{departure}$ be the time at which the robot departs it from $\boldsymbol{x}_{departure}$, Fig. 2. The objective function to maximize, then, is the time spent, searching for the target, while remaining on the iso-probability curve:

$$\text{maximize } f_{Traj} = t_{departure} - t_{interception}. \tag{8}$$

The above optimization is carried out by varying the iso-probability curve interception and departure locations, $\boldsymbol{x}_{interception}$ and $\boldsymbol{x}_{departure}$, respectively. After waypoints are optimized, trajectory segments between them would be planned to maximize the likelihood of target detection. Both $\boldsymbol{x}_{interception}$ and $\boldsymbol{x}_{departure}$ are restricted to be on iso-probability curves as the optimal search trajectory begins and ends on the iso-probability curve.

Furthermore, since, there would be multiple robots, with multiple possible corresponding iso-probability curves to assign them to, there exists the additional problem of determining optimal robot-to-curve assignments. An optimal assignment can be achieved by minimizing the variance in the search effort assigned to different iso-probability curves. For example, they can be calculated via the linear density of the number of the robots, n_c, per length of the curve, l_c, $\lambda_c = n_c/l_c$, for each curve. In this case, the objective function would be:

DOI: 10.1109/TCYB.2018.2875625
The final version of the paper is available at https://ieeexplore.ieee.org/document/8509163

$$\text{minimize } f_{Assign} = \sum_{c=1}^{N} \frac{\left(\lambda_c - \bar{\lambda}\right)^2}{N}, \tag{9}$$

where N is the total number of curves and:

$$\bar{\lambda} = \sum_{c=1}^{N} \frac{\lambda_c}{N}. \tag{10}$$

The above optimization varies robot-to-curve assignments, denoted by the binary variable a_{rc}, which is 1 if Robot r is assigned to Curve c, and 0, otherwise, to minimize Eq. (9). In the optimization, each robot can only be assigned to one curve, but multiple robots may be assigned to any single curve.

III. Hybrid Search-Planning Method

The proposed three-phase planning method first plans the time-phased deployment of the optimal static-sensor network based on available search-scenario information. Next, the optimal robot routes are planned, for making the scheduled static-sensor deliveries. Lastly, the robot-motion trajectories are planned, such that the robots actively search for the target while deploying the static-sensor network according to the planned routes. The planning method is illustrated in Fig. 3.

The overall search is planned such that the target always remains within the 'sensing field' of the searchers. Namely, the search keeps pace with the moving target to maximize the likelihood of its detection. The optimal static-sensor-network configuration and the optimal robot routes and trajectories together comprise the complete hybrid search plan.

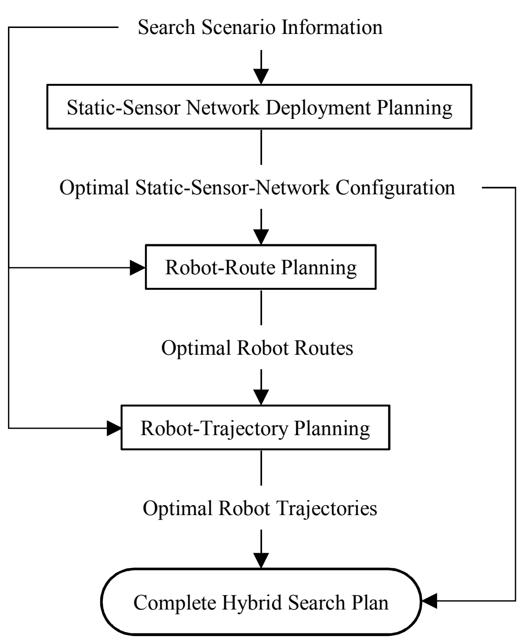

Fig. 3. The proposed hybrid search-planning method.

A. *Static-Sensor Network Deployment Planning*

The solution method presented herein for the network-planning problem is an extension of the one outlined in [35]. The proposed method first determines optimal sensor-deployment times, followed by determining the sensor-placement locations (i.e., nodes) that maximize the likelihood of target detection in a directionally unbiased manner.

Three main modifications to the original method had to be implemented: (*i*) the first sensor deployment time, t_1, is now calculated, instead of being assumed to be given, (*ii*) the possible existence of redundant sensors is considered directly in the planning of the network, and (*iii*) sensor placement locations are optimized for spatiotemporal unbiasedness rather than just spatial spread.

(i) Modification 1 (Reformulation of t_1): In the original method, t_1 was set to be the start of search time and assumed to be *a priori* known. However, when considering physical delivery, robots require time to reach their respective first sensor-deployment location from their initial location. In order to account for this reality, the first sensor deployment time can be calculated as:

$$t_1 = t_s + \delta t_m, \tag{11}$$

where t_s is the (given) start of search time, when robots can begin the search, and $\delta t_m > 0$ is the time to allow the robots to reach their first deployments *on time*. Namely, it is the longest time required for any robot to reach any potential deployment location for Sensor 1.

(ii) Modification 2 (Extended sensor-network planning): In order to provide maximum flexibility, in deploying a static-sensor network optimally, it is assumed here that the robots can be loaded with extra (redundant) sensors. Redundant sensors can facilitate optimal routing if there were to be a need to re-plan the network. Namely, since sensors cannot be transferred between robots after the search has started, the redundant sensors would allow robots to have some flexibility in the number of sensors they deploy in a reconfigured network. In order to utilize potentially available redundant search resources, we propose to deploy the set of redundant sensors only after the main network of n sensors has been deployed. The redundant sensors' deployment times can be determined by extrapolation. Namely, given the last two sensor deployment times, t_{n-1} and t_n, the deployment time for the i^{th} redundant sensor would be:

$$t_{ri} = t_n + i(t_n - t_{n-1}), i \in \{1, \cdots, n_r\}, \tag{12}$$

where n_r is the number of redundant sensors.

(iii) Modification 3 (Spatiotemporal unbiasedness): Herein, the network-deployment optimization is designed to consider temporal unbiasedness in addition to just spatial unbiasedness considered in the past. A spatiotemporally unbiased sensor deployment ensures that the sensors in the network are distributed temporally unbiased in all directions throughout the (already spatially unbiased) search. Namely, the network is 'spread' such that all directions are covered homogeneously with respect to time. Such a deployment strategy would also assist in maximizing the spare time available for robots to search for the target.

As described in Section II, spatiotemporal unbiasedness

DOI: 10.1109/TCYB.2018.2875625
The final version of the paper is available at https://ieeexplore.ieee.org/document/8509163

can be achieved by minimizing the difference between the ideal and empirical distribution of sensor-deployment angular-locations and times. The ideal distribution of sensors, $F(\theta, t)$ in Eq. (3), is a joint distribution *between* two independent variables, one of space and one of time:

$$F(\theta,t) = P(\Theta \le \theta, T \le t) = P(\Theta \le \theta)P(T \le t), \tag{13}$$

where Θ is the random variable defining the angular location of the sensor and T is the random variable for the deployment time of the sensor.

The ideal distribution of sensor deployment times, $P(T \le t)$, is the distribution of all deployment times. The ideal distribution of sensor angular locations, $P(\Theta \le \theta)$ can be determined by considering that an ideal distribution would have identical levels of coverage in all directions. Namely, sensors would ideally be uniformly distributed in the region of interest such that all directions of target travel have similar levels of coverage. Since the region of interest in this case is the area between the deployment curve for the first deployment time, t_1, and the deployment curve of the last deployment time, t_n, the ideal CDF of angular sensor positions can be defined as:

$$P(\Theta \le \theta) = \frac{A(\theta)}{A(2\pi)}, \tag{14}$$

where $A(\theta)$ is the area bounded by the initial and final deployment curves in a sector covering the angles $[0, \theta]$.

Sensor positions on curves minimizing the objective function f_{Dep_ang} in Eq. (3) could be determined through a search engine, such as particle swarm optimization [35].

An example network of 150 sensors, planned according to the modified (spatiotemporally unbiased) configuration-planning technique, is shown in Fig. 4. Blue points in the figure denote sensor-deployment positions, while the black cross represents the target LKP. The network was designed based on the demographics of a lost person, in a given search region, with known deployment-time constraints. Namely, the network was designed to optimally search for a target propagating away from the LKP with the first sensor being deployed at t_1 = 4367 s and the last sensor being deployed at t_{150} = 11,377 s, respectively. Here, t = 0 s is the time at which the target was at the LKP (the center). The center is, therefore, ‘empty’ since it is assumed that the target has already propagated some distance away from the LKP. Furthermore, static sensors are deployed progressively away from the LKP as the search advances.

The sensor network is approximately an annulus. The inner circle is defined by the time that has passed before the search starts and the outward propagation pace of the target. The outer circle is defined by the allotted total search time and the outward propagation pace of the target.

B. Robot-Route Planning

Given an optimal static-sensor network and the number of robots available and their specifications (e.g., carrying capacity and speed), the optimal robot routes need to be

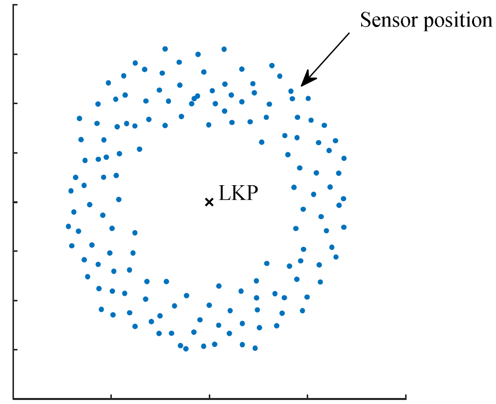

Fig. 4. An example spatiotemporally optimized sensor network.

determined. The primary goal is to maximize the (minimum) spare time over all sensor nodes, objective function f_{Route} in Eq. (6). A secondary goal of maximizing average spare time over all sensors can also be invoked, objective function f_{Route_alt} in Eq. (7). The primary and secondary goals are formulated as a two-tier objective function [59], [60]. The *max-min* objective function ensures that all sensors are deployed, at their specified deployment locations, as close as possible to their optimal deployment times. Furthermore, it ensures that the maximized spare-times are distributed in an unbiased way between robots and throughout their routes.

Robot routes are planned for the main network deployment first, while disregarding the redundant sensors. It is followed by route planning for the redundant network, with robots’ new starting points and times being defined by their respective last (main-network) sensor placements. The combinatoric problem of determining optimal routes can, for example, be solved using a Genetic Algorithm [62].

Fig. 5 shows an example set of routes planned for ten robots to deploy the network of 150 sensors shown in Fig. 4. Every robot route (i.e., a sequence of corresponding sensor deliveries) is shown with a different color. Robots start their motion at a central location (i.e., the LKP) and move outwards while deploying sensors progressively. It should be noted that the routes shown in Fig. 5 only indicate the orders (i.e., sequences) that the robots will deploy their respective sensors.

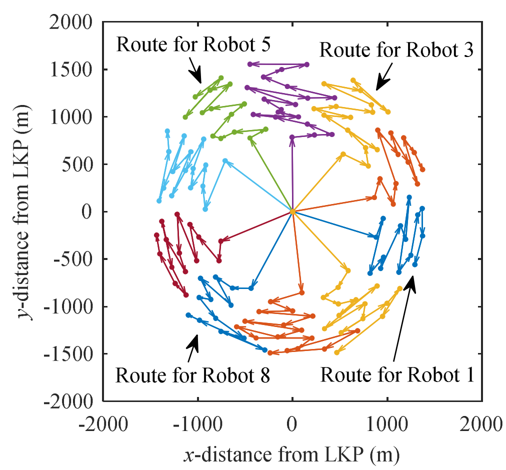

Fig. 5. Optimal delivery routes for ten robots delivering 150 sensors.

DOI: 10.1109/TCYB.2018.2875625
The final version of the paper is available at https://ieeexplore.ieee.org/document/8509163

C. *Mobile-Robot Trajectory Planning*

In an ideal target search, robots would be deployed to and remain on their respective (propagating) iso-probability curves to maximize the likelihood of target detection. An optimal search trajectory is, therefore, one that keeps a robot on its assigned iso-probability curve for as long as possible during the spare time available to it between sensor placements. Namely, the optimal trajectory is one that maximizes the time spent by the robot on the optimal search path. It should be noted that this method could be adapted to utilize any existing mobile-search path-planning algorithm.

Iso-probability curves are constructed by examining the estimated target location likelihood along rays extending from the LKP. Namely, the distribution of targets along rays are taken as estimates of the one-dimensional target-location PDF described in [61]. Equal cumulative probability points on corresponding CDFs along rays can, then, be connected to form the iso-probability curves, [61]. Fig. 6 illustrates the iso-probability curve creation process. First, Fig. 6(a) shows the target-location likelihood and an example ray (white) extending outwards from the LKP. Brighter (yellower) colors on the target-location likelihood function indicate a higher likelihood of target detection while darker (bluer) colors indicate a lower likelihood of target detection. Fig. 6(b) shows the one-dimensional target-location PDF along the example ray. Fig. 6(c) illustrates where the 10% (red) and 50% (green) points along the ray would be found. Finally, Fig. 6(d) shows the result of connecting equal cumulative probability points along multiple rays. The red curve represents the 10% iso-probability curve and the green curve represents the 50% iso-probability curve.

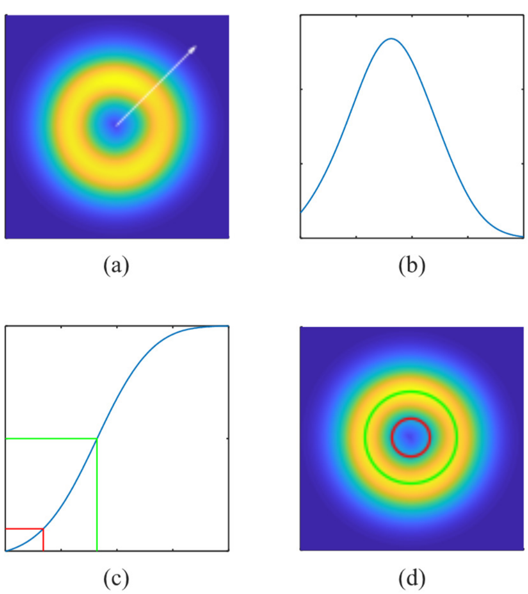

Fig. 6. The iso-probability curve creation process: (*a*) target location likelihood estimate, (*b*) a target-location PDF along the ray, (*c*) a target-location CDF with the 10% (red) and 50% (green) cumulative points indicated, and (*d*) the 10% (red) and 50% (green) iso-probability curves.

However, as abovementioned in Section II, robots are required to deploy sensors while searching for the target in the proposed method. Thus, as formulated in Section II, the robot trajectory is divided into three segments, Fig. 2. An illustrative example in which a robot moves from Sensor i to Sensor j is shown in Fig. 7. In the figure, the robot (black square) begins at Sensor i (blue dot), Fig. 7(a); it, then, intercepts its iso-probability curve (black curve), Fig. 7(b); and remains on the iso-probability curve as it is propagating with time, Figs. 7(c) and 7(d); and, finally, it leaves the curve and reaches Sensor j, Fig. 7(e). The green curve is the total robot path.

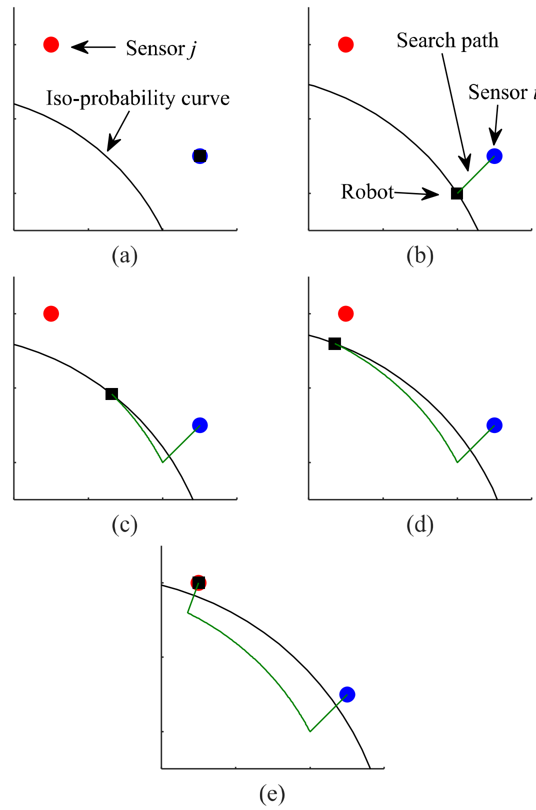

Fig. 7. An example in which a robot travels from Sensor i to Sensor j.

1) *Optimizing $\boldsymbol{x}_{interception}$*

Optimizing $\boldsymbol{x}_{interception}$ requires determining the location at which the iso-probability curve interception time, $t_{interception}$, is minimized such that the objective function f_{Traj} is maximized, Eq. (8). One may note that an iso-probability curve cannot be intercepted unless the robot can arrive at a location before the curve does, $t_{earliest\_arrival}$. Namely, a feasible curve interception location is one at which:

$$t_{earliest\_arrival} \leq t_{interception}. \tag{15}$$

The earliest robot arrival time, $t_{earliest\_arrival}$, can be found by determining the time required to traverse the fastest path from Sensor i to the location. The fastest path can be determined, for example, using a shortest path algorithm such as Dijkstra's algorithm on a graph representing discretized terrain in the search area [63]. The optimal curve-interception point, $\boldsymbol{x}_{interception}$, then, is the location at which we find the earliest feasible $t_{interception}$.

DOI: 10.1109/TCYB.2018.2875625
The final version of the paper is available at https://ieeexplore.ieee.org/document/8509163

2) Optimizing $\boldsymbol{x}_{departure}$

Optimizing $\boldsymbol{x}_{departure}$ follows a similar logic and process as the optimization of $\boldsymbol{x}_{interception}$. Namely, the curve-departure location must be optimized such that the corresponding departure time, $t_{departure}$, is such that the objective function f_{Traj} is maximized, Eq. (8).

One may note that the robot must arrive at the Sensor j deployment location in time. Namely, a feasible curve departure location is one at which:

$$t_{latest\_departure} \geq t_{departure}. \tag{16}$$

The latest robot departure time, $t_{latest\_departure}$, can be found by determining the time required to traverse the fastest path from the departure location to Sensor j. As above, the fastest path can be determined, for example, using a shortest path algorithm such as Dijkstra's algorithm [63]. The optimal curve interception point, $\boldsymbol{x}_{intercept}$, then, is the location at which we find the latest feasible $t_{departure}$.

3) Search-Trajectory Planning

The optimal robot trajectory would need the robot to intercept its iso-probability curve at $\boldsymbol{x}_{interception}$ at time $t_{interception}$ and, while remaining on it for as long as it is possible, reach $\boldsymbol{x}_{departure}$ at time $t_{departure}$, Fig. 8, below. As abovementioned, an ideal search trajectory would also cover as much of the search area as possible between sensor deployments. While there are many possible trajectories connecting $\boldsymbol{x}_{interception}$ and $\boldsymbol{x}_{departure}$, not all of them satisfy the above requirements. For example, if a robot were to move at its maximum speed in one direction along the iso-probability curve, it would, typically, overshoot $\boldsymbol{x}_{departure}$. In contrast, if it were to move at a slower speed, it could get to $\boldsymbol{x}_{departure}$, at exactly $t_{departure}$, but, its potential to search for the target between deployments would be wasted. Thus, we propose herein to allow the robot to move at its maximum speed, but, prolong its path by reversing directions as it moves on its respective iso-probability curve.

In order to achieve the above objective, we define an intermediate-goal point, labelled as a *turning point*, $\boldsymbol{x}_{turning}$, at which the robot reverses direction. Namely, the robot would first move towards $\boldsymbol{x}_{turning}$ from $\boldsymbol{x}_{interception}$, and it would, then, reverse its direction, while still remaining on its respective iso-probability curve, to head toward $\boldsymbol{x}_{departure}$, Fig. 8. The turning point can be optimized to ensure the robot arrives at $\boldsymbol{x}_{departure}$ just in time for $t_{departure}$ while achieving a maximum search path, maximizing search area coverage.

In order to plan an optimal robot trajectory, while meeting the above objective, let us assume $\boldsymbol{x}_{interception}$, $\boldsymbol{x}_{turning}$, and $\boldsymbol{x}_{departure}$ have corresponding angular positions $\theta_{interception}$, $\theta_{turning}$ and $\theta_{departure}$, respectively. Namely, after the robot travels from $\theta_{interception}$ to $\theta_{turning}$ the robot would move from $\theta_{turning}$ to $\theta_{departure}$, all while following the iso-probability curve, Fig. 8.

The radial progress of the robot as it moves angularly from $\theta_{interception}$ to $\theta_{turning}$ to $\theta_{departure}$ is defined by its angular progress. Namely, let as assume that the robot angularly moves a total distance of α_{TOT}. Then, the robot's radial progress between the search curve at $t_{interception}$ and the search curve at $t_{departure}$ is given by:

$$r(\alpha) = \frac{\alpha}{\alpha_{TOT}}. \tag{17}$$

Here, α is the total angular distance travelled up to a point on the robot path and r is the radial progress between the two search curves. For example, if the robot is halfway through its path angularly, it should also be halfway between the two search curves radially. This approximates the robot following the search curve as it propagates without having to generate the search curve at times between $t_{interception}$ and $t_{departure}$, which can be computationally prohibitive.

The above-described planning process is suitable for determining the optimal trajectory for a single robot travelling between its consecutive deployments. Typically, however, there would be multiple robots participating in the search. As such, the problem of optimally assigning robots to curves during search and deployment must be addressed. Herein, we propose to determine optimal robot-curve assignments using a Blackboard architecture [64]. An optimal assignment, thus, refers to one in which robots are distributed such that all iso-probability curves are searched, minimizing f_{Assign} in Eq. (9).

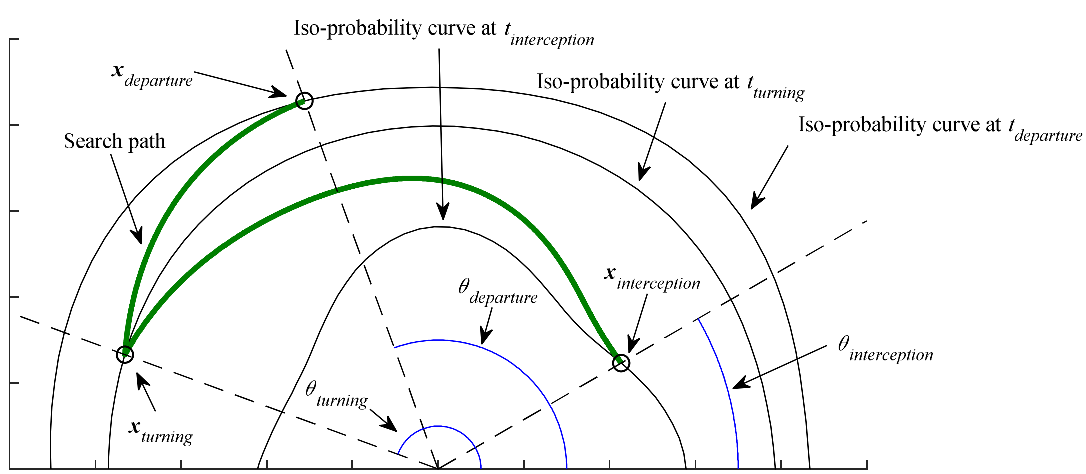

Fig. 8. An example optimal search path (green line) starting at $\boldsymbol{x}_{interception}$ and ending at $\boldsymbol{x}_{departure}$.

DOI: 10.1109/TCYB.2018.2875625
The final version of the paper is available at https://ieeexplore.ieee.org/document/8509163

D. Re-Planning

It can be noted that the hybrid-search planning process can be re-initiated at any time during the search if new information regarding target motion is to become available. For example, when a clue regarding the target's location is found during search and deployment, it would be used to update the target's LKP. Search and deployment would, then, be temporarily halted to allow for re-planning. The sensor network would be re-planned, the robots re-routed, and search trajectories re-planned to account for the new information.

This would involve first re-planning the sensor network based on the new information while also considering the already deployed set of sensors. Namely, ensuring there are no redundant sensor deployments. One can note that, during sensor-network planning, the locations of the redundant sensors relative to the new LKP would need to be translated to corresponding spatiotemporal coordinates to be considered in the optimization. Namely, it is necessary to compute which deployment curve the sensor would be deployed on to determine the temporal component and what angular position it is relative to the new LKP to determine the spatial component. Further details on sensor-network re-planning can be found in [35].

Next, robot routes would be re-planned, with robots beginning where they halted (at the time of re-planning initiation). Route re-planning would be carried out as described in Section III.B. However, the additional constraint of robot sensor-deployment capacity would be added to route optimization. There was no capacity constraint in the original route planning since any number of sensors could be loaded on to the robots prior to the start of the search. During re-planning, however, sensors cannot be re-distributed among the robots. Namely, each robot can only deploy up to a finite number of sensors.

Finally, robot search trajectories would be re-planned based on the planned routes. The process is identical to that described in Section III.C. Paths are planned to maximize the likelihood of target detection based on the re-estimated target motion and a new set of iso-probability curves.

IV. Deployment Examples

The proposed hybrid search-planning method was validated by simulating a variety of search scenarios. Two comprehensive examples are first detailed in Sub-sections IV.A and IV.B. Sub-section IV.C, in turn, presents an analysis of the performance of the proposed hybrid method.

A. Search Example 1

In this first WiSAR example scenario, the target walking speed was assumed to be distributed according to (μ = 0.24 m/s, σ = 0.08 m/s), where the target motion model was simulated using $\sigma_\theta = \pi/3$ rad and d_m = 100 m. σ_θ represents the degree to which the target wanders and d_m represents the maximum distance for which the target will maintain a given heading as detailed in [35]. The search area, with varying terrain, shown in Fig. 9, was used for the examples. In the figure, darker colors denote higher elevation and solid orange-colored shapes denote impassable obstacles.

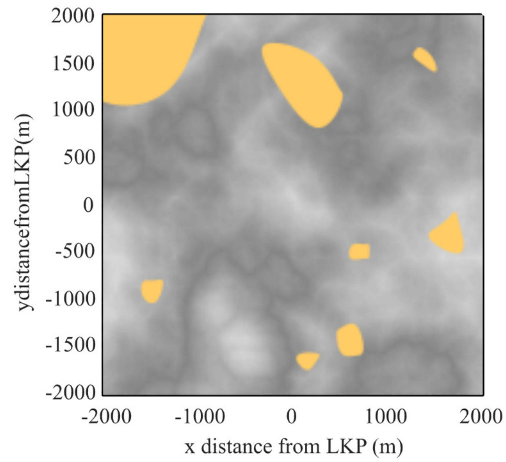

Fig. 9. Terrain of the search area experiments.

The resources available for the search included 150 static sensors, 135 sensors for the main network and 15 redundant sensors. Ten robots capable of moving at up to 2.4 m/s were available as mobile agents. The static sensors were assumed to have a sensing radius of 60 m, while (mobile) sensors onboard the robots had a sensing radius of 20 m.

It was assumed that the search begins at 2,000 s after the distress call. Consequently, the placement time of the first sensor was calculated to be t_1 = 2,321 s. The placement of the last sensor in the main network, t_{135}, was set to 16,000 s, and the overall search end time, t_e, was set to 20,000 s.

Fig. 10 shows the complete planned 150-sensor network, per the method described in Section III.A, with sensors shown as blue points. Based on optimal routes planned, the robots had a minimum spare time of about 220 s to search for the target between sensor deployments, with an average spare time of about 930 s.

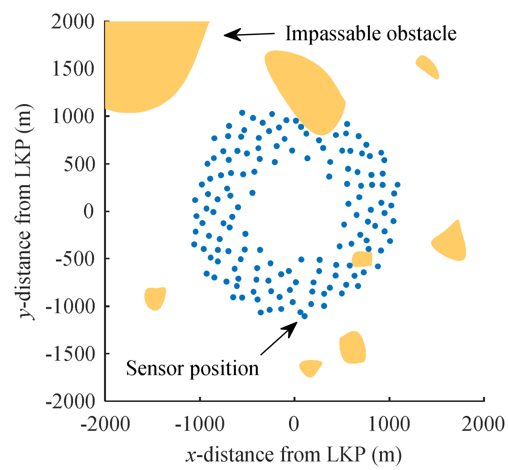

Fig. 10. The planned static-sensor network for Example 1.

The optimal network-deployment and robot-routing plan was, then, tested for numerous simulated target-motion scenarios, two of which are illustrated in Figs. 11 and 12, respectively. In the two scenarios, the search plan was identical, but the target being sought took a different trajectory. In the first scenario, the target was detected by a

DOI: 10.1109/TCYB.2018.2875625
The final version of the paper is available at https://ieeexplore.ieee.org/document/8509163

sensor while a robot detected the target in the second scenario.

Fig. 11 shows a snapshot of the end of search scenario 1, where the target (green ×) is found at t = 5,307 s by Sensor #3 (red dot), that was deployed earlier by Robot #2 at time t = 2,646 s. In this figure, the target path is shown by a purple-colored line, the already deployed 22 sensors are shown by blue dots (except for Sensor #3 shown in red), and all robots are shown on black squares.

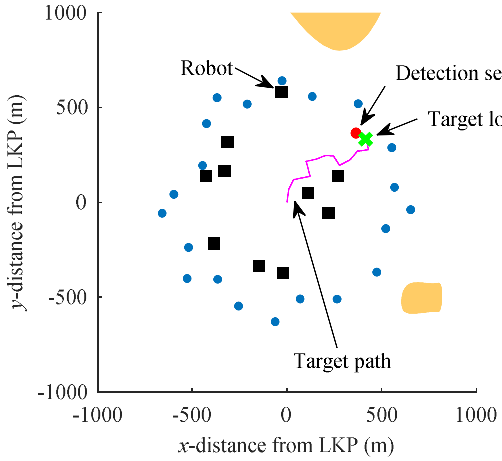

Fig. 11. End of search Scenario 1 in Example 1.

Fig. 12 shows a snapshot of the end of search scenario 2, where the target is found at t = 4,678 s by Robot #1. The robot was on its way from having deployed Sensor #8 moving toward Sensor #20 location.

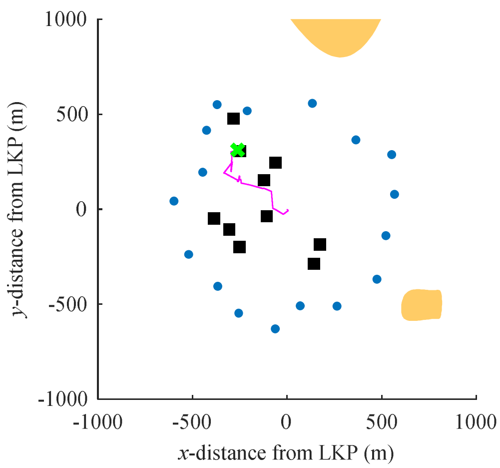

Fig. 12. End of search Scenario 2 in Example 1.

B. Search Example 2

In this example, demographics information was used to infer that the target would likely have a walking speed distributed according to (μ = 1.0 m/s, σ = 0.33 m/s). Furthermore, a target motion model with $\sigma_\theta = \pi/2$ rad and d_m = 50 m was assumed. The search began 1000 s after the target was known to be at the LKP and was planned to continue until t = 10,000 s. A total of 100 sensors (80 main, 20 redundant) were deployed between t_1 = 1221 s and t_{100} = 6,985 s by 15 robots. It was assumed that the robots can travel at 10 m/s and had a sensing radius of 60 m. Static sensors were assumed to have a sensing radius of 20 m.

Fig. 13 shows the planned sensor network with robot routes. Route optimization provided robots with a minimum spare time of 121 s and a mean spare time of 804 s.

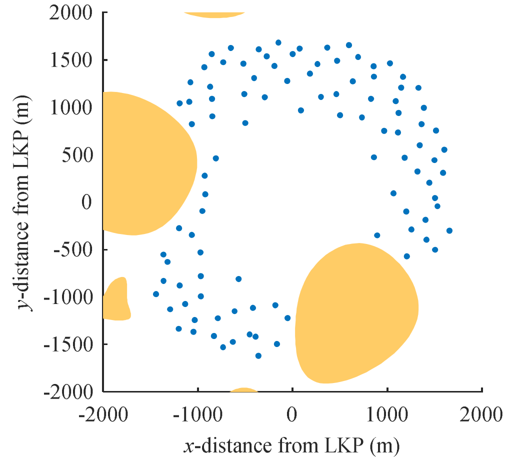

Fig. 13. The planned static-sensor network for Example 2.

Fig. 14 shows a snapshot of the end of a search where the target was detected by mobile-robot #7 on its way between Sensors #70 and #82. This target was found at t = 6003 s.

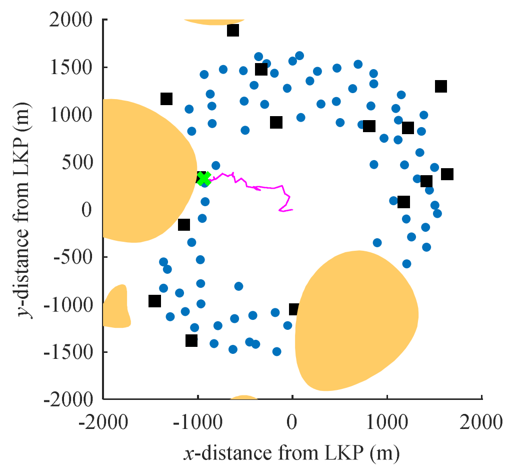

Fig. 14. End of search scenario in Example 2.

C. Method-Performance Analysis

In order to analyze the performance of the proposed search-planning method, numerous searches were planned and 1,000 different target motions were simulated for each search plan. Search plans considered various combinations of resource quality (e.g., detection radius) and quantity (e.g., number of sensors and robots). Different sensor detection radii, ranging between 2 m and 100 m, were investigated while mobile robot-team sizes investigated ranged from 8 to 20 robots and sensor-network sizes ranged from 50 to 300 sensors.

One may note that past work has established the effectiveness of the mobile-robot search, static-sensor network search, and route-planning individually. Namely, the work in [21] demonstrated the superiority of the proposed path planning approach when compared to a more typical grid-search like path planning approach. The work in [35] compared a preliminary version of the static-sensor network deployment method to a uniform coverage method. The

DOI: 10.1109/TCYB.2018.2875625
The final version of the paper is available at https://ieeexplore.ieee.org/document/8509163

comparison showed that the proposed method significantly improves both target detection likelihood and mean detection time over the uniform coverage approach. The work in [60] similarly compared the proposed route-planning method to multiple well-known alternatives such as genetic algorithms, ant colony optimization, and simulated annealing. The comparisons showed that the proposed method resulted in significantly better routes given the same amount of computation time. Since no work has yet considered a hybrid approach, like the one proposed in this paper, the analysis below compares the performance of the proposed hybrid search to that of a static-sensor only search.

1) Sensor-Network-Delivery Planning Performance

Planning performance was evaluated in terms of the number of sensors deployed on-time and the spare time available to robots during sensor deployment. A summary of four representative search plans are presented below in Table I. As can be noted, a lower sensor/robot ratio results in a higher likelihood of sensors being deployed on-time, i.e., with more spare time between sensor deployments.

When the sensor/robot ratio is lower, each robot would have fewer sensors to deploy. This results in the average time between deployments to be larger. As robots would have more time between deployments, they would have a higher likelihood of deploying the sensors on-time as well as more spare time to search for the target.

TABLE I: SENSOR NETWORKS

Number of: Sensors / Robots	***Number of Late Sensors: Main / Redundant / Total***	***Spare Time Min / Mean / Max (s)***
50 / 10	0 / 0 / 0	96 / 1,235 / 3,097
100 / 10	0 / 0 / 0	52 / 572 / 2,539
150 / 10	9 / 2 / 11	–85 / 351/ 1,159
200 / 10	33 / 8 / 41	–102 / 225 / 877

2) Overall Search Performance Comparison

The overall search performance of the proposed search method was compared to that of a search carried out solely by a static-sensor network. Namely, 1,000 simulated searches were performed on 1,000 different targets according to plans made by the two methods for each of the four sensor network sizes. The results of those searches were, then, compared to investigate the improvement in search performance when using the proposed hybrid approach. The performance of the searches was evaluated by computing target detection rates (i.e., how many of the 1,000 simulated searches were successful) and mean detection times (i.e., mean time to target detection over the 1,000 searches). In each simulated search, targets moved until discovered by either a static sensor or a mobile sensor on a robot or until the end of search time.

Tables II below shows results wherein the static sensors had a sensing radius of 60 m and the robots had a sensing radius of 20 m. Table III shows results wherein the static sensors had a sensing radius of 20 m and the robots had a sensing radius of 60 m. As in Table I, the first column shows the four sensor-robot ratios considered. The second column shows the total number of targets found in each hybrid-search (HS) case. The third column shows the number of targets found by the static-sensor network (SN). The fourth column indicates the amount of coverage provided by the sensor network and the fifth column indicates the hybrid-search (HS) improvement over the static search (SN).

The results indicate that utilizing mobile agents to search for the target while deploying the static-sensor network can tangibly improve target detection rates over a static-sensor network only search. Improvements above 400% were observed, where the smaller the static-sensor network coverage of the search area is, the better the improvement. As the static-sensor network grows larger, the improvement falls with robots detecting less targets. This trend was observed for both cases in which the static-sensors had a larger sensing radius and in cases where mobile-robots had a larger sensing radius. The trend indicates that, for large networks, the robots detect targets first, which would have been detected later by the static-sensor network. However, as expected, it was noted that, adding a robot team to support the static-sensor network does not lead to detecting targets earlier.

The trend of diminishing returns also suggests that there is an optimal hybridization problem that could be addressed in future work. While not relevant to this work, wherein improvement in search performance is considered desirable at any cost, there could be a cost-performance optimization problem that is relevant to less critical applications.

TABLE II: DETECTION RATE PERFORMANCE FOR STATIC 60 - MOBILE 20 SCENARIOS

# of: Sensors / Robots	***# Targets Found by HS***	***# Targets Found by SN***	***Coverage by Sensors (%)***	***Improvement of HS over SN (%)***
200 / 10	705	676	35	4
150 / 10	699	634	26	10
100 / 10	687	567	18	21
50 / 10	637	396	9	61

TABLE III: DETECTION RATE PERFORMANCE FOR STATIC 20 - MOBILE 60 SCENARIOS

# of: Sensors / Robots	***# Targets Found by HS***	***# Targets Found by SS***	***Coverage by Sensors (%)***	***Improvement of HS over SS (%)***
200 / 10	745	426	4	75
150 / 10	728	366	3	99
100 / 10	741	269	2	175
50 / 10	754	147	1	413

V. CONCLUSIONS

This paper presents a novel hybrid target-search strategy, where a static-sensor network is supported by a mobile search effort. The corresponding proposed planning method first determines a spatiotemporally optimized static-sensor network deployment plan to facilitate physical deployment and maximize the likelihood of target detection. Thereafter, robot routes and trajectories are planned to deliver the sensors to optimal static-sensor locations *just-in-time*.

The strategy is novel in that robots that deploy sensors also

DOI: 10.1109/TCYB.2018.2875625
The final version of the paper is available at https://ieeexplore.ieee.org/document/8509163

actively search for the target between deliveries. This is in contrast to most other mobile target-search methods that, typically, only use robots for one purpose at a time (e.g., robots only search for the target after sensor deployment is completed). Additionally, the method uniquely plans sensor networks to facilitate physical delivery to improve the ability of robots to search between deliveries.

The proposed method was illustrated via simulated WiSAR experiments. Simulations were also performed to investigate the effect of mobile and static search resource availability on planning and search performance. Results showed that using robots to search, while deploying a static-senor network, could tangibly improve success in locating the mobile-target.

While the method presented herein was, primarily, developed for real-time WiSAR, it could be applied to any mobile-target search problem in which a target-motion model is available and a target-location likelihood function can be estimated. Also, although this paper focuses on the overall search-planning method, it does not propose the use of any specific optimization search engine. Thus, future work could include an investigation for determining a suitable search engine to efficiently optimize sensor-network configurations and robot trajectories.

References

[1] J. Berger and N. Lo, "An innovative multi-agent search-and-rescue path planning approach," *Comput. Oper. Res.*, vol. 53, pp. 24–31, Jan. 2015.

[2] P. J. Doherty, Q. Guo, J. Doke, and D. Ferguson, "An analysis of probability of area techniques for missing persons in Yosemite National Park," *Appl. Geogr.*, vol. 47, pp. 99–110, Feb. 2014.

[3] M. Israel, E. Khmelnitsky, and E. Kagan, "Search for a mobile target by ground vehicle on a topographic terrain," in *IEEE Conv. Elect. Electron. Eng. Israel*, Eilat, Israel, 2012, pp. 1–5.

[4] A. Macwan, G. Nejat, and B. Benhabib, "Optimal deployment of robotic teams for autonomous wilderness search and rescue," in *IEEE/RSJ Int. Conf. Intell. Robots Syst.*, San Francisco, CA, 2011, pp. 4544–4549.

[5] E.-M. Wong, F. Bourgault, and T. Furukawa, "Multi-vehicle Bayesian Search for Multiple Lost Targets," in *Int. Conf. Robotics Automation*, Barcelona, Spain, 2005, pp. 3169–3174.

[6] L. Champagne, E. G. Carl, and R. Hill, "Search theory, agent-based simulation, and u-boats in the Bay of Biscay," in *Simulation Conf.*, New Orleans, LA, 2003, vol. 1, p. 991–998 Vol.1.

[7] A. Arora *et al.*, "A Line in the Sand: A Wireless Sensor Network for Target Detection, Classification, and Tracking," *Comput. Netw.*, vol. 46, no. 5, pp. 605–634, Dec. 2004.

[8] A. Kehagias, D. Mitsche, and P. Prałat, "Cops and invisible robbers: The cost of drunkenness," *Theor. Comput. Sci.*, vol. 481, pp. 100–120, Apr. 2013.

[9] L. Lin and M. A. Goodrich, "Hierarchical Heuristic Search Using a Gaussian Mixture Model for UAV Coverage Planning," *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2532–2544, Dec. 2014.

[10] W. Zhao, Q. Meng, and P. W. H. Chung, "A Heuristic Distributed Task Allocation Method for Multivehicle Multitask Problems and Its Application to Search and Rescue Scenario," *IEEE Trans. Cybern.*, vol. 46, no. 4, pp. 902–915, Apr. 2016.

[11] C. M. Keller, "Applying optimal search theory to inland SAR: Steve Fossett case study," in *Conf. Inform. Fusion*, Endinburgh, UK, 2010, pp. 1–8.

[12] R. Hohzaki, "A cooperative game in search theory," *Nav. Res. Logist.*, vol. 56, no. 3, pp. 264–278, Apr. 2009.

[13] S. J. Benkoski, M. G. Monticino, and J. R. Weisinger, "A survey of the search theory literature," *Nav. Res. Logist.*, vol. 38, no. 4, pp. 469–494, Aug. 1991.

[14] D. A. Grundel, "Searching for a moving target: optimal path planning," in *Proc. Networking, Sensing and Control*, Tucson, AZ, 2005, pp. 867–872.

[15] M. A. Goodrich, B. S. Morse, C. Engh, J. L. Cooper, and J. A. Adams, "Towards using Unmanned Aerial Vehicles (UAVs) in Wilderness Search and Rescue: Lessons from field trials," *Interact. Stud.*, vol. 10, no. 3, pp. 453–478, Dec. 2009.

[16] Y. F. Ding and Q. Pan, "Path Planning for Mobile Robot Search and Rescue Based on Improved Ant Colony Optimization Algorithm," *Appl. Mech. Mater.*, vol. 66–68, pp. 1039–1044, Jul. 2011.

[17] H. Xiao, R. Cui, and D. Xu, "A Sampling-Based Bayesian Approach for Cooperative Multiagent Online Search With Resource Constraints," *IEEE Trans. Cybern.*, vol. PP, no. 99, pp. 1–13, 2017.

[18] D. J. Pack, P. DeLima, G. J. Toussaint, and G. York, "Cooperative Control of UAVs for Localization of Intermittently Emitting Mobile Targets," *IEEE Trans. Syst. Man Cybern. Part B Cybern.*, vol. 39, no. 4, pp. 959–970, Aug. 2009.

[19] M. T. Agcayazi, E. Cawi, A. Jurgenson, P. Ghassemi, and G. Cook, "ResQuad: Toward a semi-autonomous wilderness search and rescue unmanned aerial system," in *Int. Conf. Unmanned Aircraft Syst.*, Arlington, TX, 2016, pp. 898–904.

[20] A. Macwan, J. Vilela, G. Nejat, and B. Benhabib, "Multi-Robot Deployment for Wilderness Search and Rescue," *Int. J. Robot. Autom.*, vol. 31, no. 1, 2016.

[21] A. Macwan, J. Vilela, G. Nejat, and B. Benhabib, "A Multirobot Path-Planning Strategy for Autonomous Wilderness Search and Rescue," *IEEE Trans. Cybern.*, vol. 45, no. 9, pp. 1784–1797, Sep. 2015.

[22] J. A. Rogge and D. Aeyels, "Multi-robot coverage to locate fixed and moving targets," in *Control Appl., Intell. Control*, St. Petersburg, Russia, 2009, pp. 902–907.

[23] B. Lavis, T. Furukawa, and H. F. Durrant Whyte, "Dynamic space reconfiguration for Bayesian search and tracking with moving targets," *Auton. Robots*, vol. 24, no. 4, pp. 387–399, Jan. 2008.

[24] Y. Sung and T. Furukawa, "Information measure for the optimal control of target searching via the grid-based method," in *Int. Conf. Inform. Fusion*, Heidelberg, Germany, 2016, pp. 2075–2080.

[25] H. Yuan *et al.*, "Target Detection, Positioning and Tracking Using New UAV Gas Sensor Systems: Simulation and Analysis," *J. Intell. Robot. Syst.*, pp. 1–12, Jul. 2018. doi: 10.1007/s10846-018-0909-2

[26] D. Hanna, A. Ferworn, M. Lukaczyn, A. Abhari, and J. Lum, "Using Unmanned Aerial Vehicles (UAVs) in locating wandering patients with dementia," in *IEEE/ION Position, Location Navigation Symposium*, Monterey, CA, 2018, pp. 809–815.

[27] E. Amaldi, A. Capone, M. Cesana, and I. Filippini, "Design of Wireless Sensor Networks for Mobile Target Detection," *IEEEACM Trans. Netw.*, vol. 20, no. 3, pp. 784–797, Jun. 2012.

[28] T. Clouqueur, V. Phipatanasuphorn, P. Ramanathan, and K. K. Saluja, "Sensor Deployment Strategy for Detection of Targets Traversing a Region," *Mob. Netw. Appl.*, vol. 8, no. 4, pp. 453–461, Aug. 2003.

[29] V. Phipatanasuphorn and P. Ramanathan, "Vulnerability of sensor networks to unauthorized traversal and monitoring," *IEEE Trans. Comput.*, vol. 53, no. 3, pp. 364–369, Mar. 2004.

[30] L. Lazos, R. Poovendran, and J. A. Ritcey, "Detection of mobile targets on the plane and in space using heterogeneous sensor networks," *Wirel. Netw.*, vol. 15, no. 5, pp. 667–690, Jul. 2009.

[31] Y. Yoon and Y.-H. Kim, "An Efficient Genetic Algorithm for Maximum Coverage Deployment in Wireless Sensor Networks," *IEEE Trans. Cybern.*, vol. 43, no. 5, pp. 1473–1483, Oct. 2013.

[32] K. Mukherjee, S. Gupta, A. Ray, and T. A. Wettergren, "Statistical-Mechanics-Inspired Optimization of Sensor Field Configuration for Detection of Mobile Targets," *IEEE Trans. Syst. Man Cybern. Part B Cybern.*, vol. 41, no. 3, pp. 783–791, Jun. 2011.

[33] M. Karatas, E. Craparo, and G. Akman, "Bistatic sonobuoy deployment strategies for detecting stationary and mobile underwater targets," *Nav. Res. Logist. NRL*, 2018. doi: 10.1002/nav.21807

[34] J. Vilela, Z. Kashino, R. Ly, G. Nejat, and B. Benhabib, "A Dynamic Approach to Sensor Network Deployment for Mobile-Target Detection in Unstructured, Expanding Search Areas," *IEEE Sens. J.*, vol. 16, no. 11, pp. 4405–4417, Jun. 2016.

[35] Z. Kashino, J. Y. Kim, G. Nejat, and B. Benhabib, "Spatiotemporal Adaptive Optimization of a Static-Sensor Network via a Non-Parametric Estimation of Target Location Likelihood," *IEEE Sens. J.*, vol. 17, no. 5, pp. 1479–1492, Mar. 2017.

DOI: 10.1109/TCYB.2018.2875625
The final version of the paper is available at https://ieeexplore.ieee.org/document/8509163

[36] S. Shue and J. M. Conrad, "A survey of robotic applications in wireless sensor networks," in *Proc. IEEE Southeastcon*, Jacksonville, FL, 2013, pp. 1–5.

[37] T. P. Lambrou, C. G. Panayiotou, S. Felici, and B. Beferull, "Exploiting Mobility for Efficient Coverage in Sparse Wireless Sensor Networks," *Wirel. Pers. Commun.*, vol. 54, no. 1, pp. 187–201, Apr. 2009.

[38] T. Suzuki, R. Sugizaki, K. Kawabata, Y. Hada, and Y. Tobe, "Autonomous Deployment and Restoration of Sensor Network using Mobile Robots," *Int. J. Adv. Robot. Syst.*, p. 1, 2010.

[39] X. Li, G. Fletcher, A. Nayak, and I. Stojmenovic, "Randomized carrier-based sensor relocation in wireless sensor and robot networks," *Ad Hoc Netw.*, vol. 11, no. 7, pp. 1951–1962, Sep. 2013.

[40] X. Li, R. Falcon, A. Nayak, and I. Stojmenovic, "Servicing wireless sensor networks by mobile robots," *IEEE Commun. Mag.*, vol. 50, no. 7, pp. 147–154, Jul. 2012.

[41] Y. Wang and C. H. Wu, "Robot-Assisted Sensor Network Deployment and Data Collection," in *Int. Symp. Comput. Intell. Robotics Automation*, Jacksonville, FL, 2007, pp. 467–472.

[42] N. Deshpande, E. Grant, and T. C. Henderson, "Target Localization and Autonomous Navigation Using Wireless Sensor Networks - A Pseudogradient Algorithm Approach," *IEEE Syst. J.*, vol. 8, no. 1, pp. 93–103, Mar. 2014.

[43] L. Lin and M. A. Goodrich, "A Bayesian approach to modeling lost person behaviors based on terrain features in Wilderness Search and Rescue," *Comput. Math. Organ. Theory*, vol. 16, no. 3, pp. 300–323, Jul. 2010.

[44] S. Hayat, E. Yanmaz, T. X. Brown, and C. Bettstetter, "Multi-objective UAV path planning for search and rescue," in *IEEE Int. Conf. Robotics Automation*, Singapore, Singapore, 2017, pp. 5569–5574.

[45] T. Niedzielski *et al.*, "A real-time field experiment on search and rescue operations assisted by unmanned aerial vehicles," *J. Field Robot.*, Mar. 2018.

[46] A. Bakhtari, M. D. Naish, M. Eskandari, E. A. Croft, and B. Benhabib, "Active-vision-based multisensor surveillance - an implementation," *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.*, vol. 36, no. 5, pp. 668–680, Sep. 2006.

[47] F. Körner, R. Speck, A. H. Göktogan, and S. Sukkarieh, "Autonomous airborne wildlife tracking using radio signal strength," in *IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Taipei, Taiwan, 2010, pp. 107–112.

[48] T. H. Chung, G. A. Hollinger, and V. Isler, "Search and pursuit-evasion in mobile robotics," *Auton. Robots*, vol. 31, no. 4, pp. 299–316, Jul. 2011.

[49] J. Zheng, H. Yu, W. Liang, and P. Zeng, "Probabilistic strategies to coordinate multiple robotic pursuers in pursuit-evasion games," in *IEEE Int. Conf. Robotics Biomimetics,* Sanya, China, 2007, pp. 559–564.

[50] Y. Liu and G. Nejat, "Multirobot Cooperative Learning for Semiautonomous Control in Urban Search and Rescue Applications," *J. Field Robot.*, vol. 33, no. 4, pp. 512–536, Jun. 2016.

[51] J. P. Ramirez-Paredes, E. A. Doucette, J. W. Curtis, and N. R. Gans, "Urban target search and tracking using a UAV and unattended ground sensors," in *American Control Conf.*, Chicago, IL, 2015, pp. 2401–2407.

[52] H. Yu, K. Meier, M. Argyle, and R. W. Beard, "Cooperative Path Planning for Target Tracking in Urban Environments Using Unmanned Air and Ground Vehicles," *IEEE/ASME Trans. Mechatron.*, vol. 20, no. 2, pp. 541–552, Apr. 2015.

[53] Z. Zhang, G. Nejat, H. Guo, and P. Huang, "A novel 3D sensory system for robot-assisted mapping of cluttered urban search and rescue environments," *Intell. Serv. Robot.*, vol. 4, no. 2, pp. 119–134, Apr. 2011.

[54] A. Hong, O. Igharoro, Y. Liu, F. Niroui, G. Nejat, and B. Benhabib, "Investigating Human-Robot Teams for Learning-Based Semi-autonomous Control in Urban Search and Rescue Environments," *J. Intell. Robot. Syst.*, Aug. 2018. doi: 10.1007/s10846-018-0899-0

[55] R. J. Koester, *Lost Person Behavior: A Search and Rescue Guide on Where to Look for Land, Air and Water*. Charlottesville, VA: dbS Productions, 2008.

[56] C. D. Heth and E. H. Cornell, "Characteristics of Travel by Persons Lost in Albertan Wilderness Areas," *J. Environ. Psychol.*, vol. 18, no. 3, pp. 223–235, Sep. 1998.

[57] W. Mohibullah and S. J. Julier, "Stigmergic search for a lost target in wilderness," in *Sensor Signal Processing Defence*, London, UK, 2011, pp. 1–5.

[58] E. Sava, C. Twardy, R. Koester, and M. Sonwalkar, "Evaluating Lost Person Behavior Models," *Trans. GIS*, vol. 20, no. 1, pp. 38–53, Feb. 2016.

[59] Z. Kashino, G. Nejat, and B. Benhabib, "A multi-robot sensor-delivery planning strategy for static-sensor networks," in *IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Vancouver, Canada, 2017, pp. 6640–6647.

[60] K. Woiceshyn, Z. Kashino, G. Nejat, and B. Benhabib, "Vehicle Routing for Resource Management in Time-Phased Deployment of Sensor Networks," *IEEE Trans. Autom. Sci. Eng.*, 2018. doi: 10.1109/TASE.2018.2857630

[61] A. Macwan, G. Nejat, and B. Benhabib, "Target-Motion Prediction for Robotic Search and Rescue in Wilderness Environments," *IEEE Trans. Syst. Man Cybern. Part B Cybern.*, vol. 41, no. 5, pp. 1287–1298, Oct. 2011.

[62] P. Larrañaga, C. M. H. Kuijpers, R. H. Murga, I. Inza, and S. Dizdarevic, "Genetic Algorithms for the Travelling Salesman Problem: A Review of Representations and Operators," *Artif. Intell. Rev.*, vol. 13, no. 2, pp. 129–170, Apr. 1999.

[63] Z. Magyari-Sáska and Ş. Dombay, "Determining minimum hiking time using DEM," *Geogr. Napoc. Anul*, vol. 82, no. 4, pp. 124–129, 2012.

[64] J. Dong, S. Chen, and J.-J. Jeng, "Event-based blackboard architecture for multi-agent systems," in *Int. Conf. Inform. Technol.: Coding Comput.*, Las Vegas, NV, 2005, vol. 2, pp. 379–384.