Minimum Time Multi-UGV Surveillance [⋆]

David Anisi[1] and Petter Ögren[2]

[1] Optimization and Systems Theory,
Royal Institute of Technology (KTH),
SE-100 44 Stockholm, Sweden,
anisi@kth.se
[2] Department of Autonomous Systems,
Swedish Defence Research Institute (FOI),
SE-164 90 Stockholm, Sweden,
petter.ogren@foi.se

Abstract. This chapter addresses the problem of concurrent task and path planning for a number of surveillance Unmanned Ground Vehicles (UGVs) such that a user defined area of interest is covered by the UGVs' sensors in minimum time.

We first formulate the problem, and show that it is in fact a generalization of the Multiple Traveling Salesmen Problem (MTSP), which is known to be $\mathcal{NP}$-hard. We then propose a solution that decomposes the problem into three subproblems. The first is to find a maximal convex covering of the search area. Most results on static coverage use disjoint partitions of the search area, *e.g.*, triangulation, to convert the continuous sensor positioning problem into a discrete one. However, by a simple example, we show that a highly overlapping set of maximal convex sets is better suited for minimum time coverage.

The second subproblem is a combinatorial assignment and ordering of the sets in the cover. Since the Tabu search algorithm is known to perform well on various routing problems, we use it as a part of our proposed solution.

Finally, the third subproblem utilizes a particular shortest path subroutine in order to find the vehicle paths, and calculate the overall objective function used in the Tabu search. The proposed algorithm is illustrated by a number of simulation examples.

1 Introduction

Surveillance is an application area that has received an increasing amount of attention over the last decades. In civilian as well as military applications, automated solutions ranging from security cameras to surveillance UGVs are used in increasing numbers. It is therefore not surprising that the research area of automated positioning and control of surveillance sensors is also active and

[⋆] Funded by the Swedish defence materiel administration (FMV) and the Swedish armed forces through the Technologies for Autonomous and Intelligent Systems (TAIS) project. 297316-LB704859

growing. In this chapter we investigate how small scale UGVs, such as the one depicted in Figure 1, can be used in surveillance and security applications.

For the purpose of this chapter, we divide the rich set of work in this field into the following three categories: First moving sensor platforms where the main limitation on field of view is the physical sensor range. Applications where such a formulation is reasonable include demining, vacuum cleaning, UAV-search and outdoor pursuit-evasion games. The second category consists of problems dealing with positioning of static sensors where occluding objects and walls present the main limitation on field of view. Such formulations are found in the so called *Art Gallery Problems*, where the number of guards required to monitor a building is to be minimized. The third category consists of problems where moving sensors are to cover an area where again, occluding objects and walls present the main limitation of field of view. This last category includes applications such as pursuit-evasion games or exploration and mapping, in urban or indoor environments. We will now discuss each of these categories in more detail.

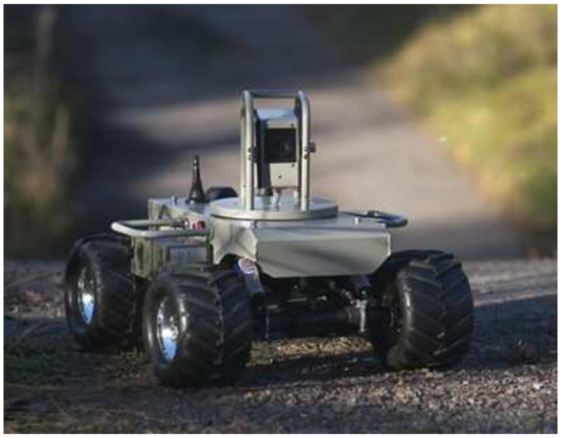


Fig. 1. This Surveillance UGV testbed, developed by SAAB Aerotech, will be used in real-world experiments.

In the first category, where sensor range is the main limitation on the extension of the visible area we find problems such as vacuum cleaning and demining, [1, 2], general coverage [3–5], multi robot coverage [6] and some robotic security applications [7]. Furthermore, a number of UAV surveillance papers, such as [8–10] fall into this category. The last set of papers also consider the combined problems of ordering a set of surveillance areas, and planning the search sweep of each individual area. The pursuit-evasion problem, where a number of pursuers try to find an evader is sometimes also formulated in this way [11].

In the second category, the field of view of stationary sensors is limited by occluding objects instead of physical sensor range. This corresponds to indoor or urban environments, where the distance between *e.g.*, walls, is in general smaller than the range of the sensor, *e.g.*, a camera or a laser scanner. The first group of results in this category comes from combinatorial geometry, and addresses Art Gallery Problems, see *e.g.*, [12,13] and the excellent survey in [14]. This work has then been built upon in [15] where a feedback solution to the guard positioning was proposed.

In the third category, where the field of view of moving guards is mainly limited by occluding walls and other objects, we also find results building on the Art Gallery work. In an indoor environment, the pursuit-evasion problem can be solved with a guarantee that the evader will be caught. Such results are found in [16, 17]. Some of this work also deals with the situation where the area is unknown. These problems are sometimes referred to as exploration and mapping, and examples include [18, 19].

Some papers address coverage problems that do not fall into one of the above categories. Examples include [20], where the mean squared distance from a sensor to a random event is minimized, and [21], where both sensor range and occlusions are incorporated into a combined planning of both UAV and sensor movements.

In the first category, many papers study the problem of covering an area in minimum time. However, when occlusions and not sensor range are the main limitations to field of view, as in the third category, we have found no paper addressing the minimum time coverage problem. In this chapter we formulate such a problem and propose an algorithm to solve it.

The organization of this chapter is as follows. In Section 2 some concepts and results from combinatorial geometry and multi vehicle routing problems is given. Then, in Section 3, we state our problem and propose a solution in Section 4. Simulations illustrating the approach are presented in Section 5. Finally, the chapter is concluded in Section 6.

2 Theoretical Background

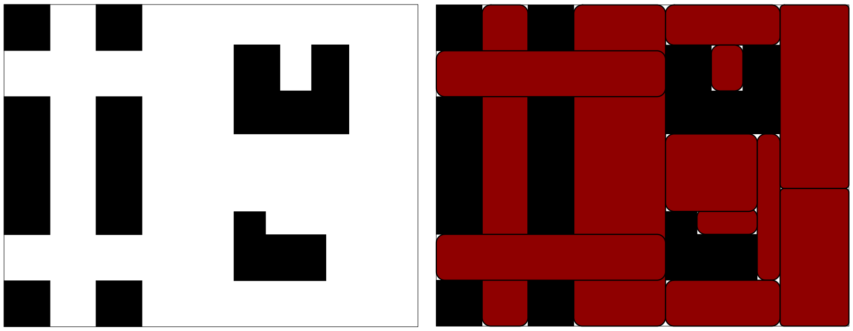
In this section we review some tools from combinatorial geometry and combinatorial optimization that will be used in the rest of the chapter.

2.1 Art Gallery Problems

We start by reviewing some terminology from combinatorial geometry. In this section we closely follow the approach of Urrutia [14], but add the sensor range R into some of the definitions.

A *polygon* Q in the plane is an ordered sequence of points $q_1, \ldots, q_n \in \mathbb{R}^2,\ n \geq 3$, called vertices of Q together with the line segments q_i to q_{i+1} for $i = 1, \ldots, n-1$ and q_n to q_1, called edges. In the following we assume that none of these edges intersect. A polygon is called *orthogonal* if adjacent edges are orthogonal.

Given a polygon Q and and a set of m disjoint polygons $Q_1, \ldots, Q_m$ contained in Q we call the set $A = Q \setminus \{Q_1 \cup ... \cup Q_m\}$ a *polygon with* m *holes*. A polygon with holes is called *orthogonal* if every pair of edges are either orthogonal or



(a) A polygon with holes (b) Convex Cover of the polygon.

Fig. 2.

parallel. The areas to be searched in this chapter, denoted A, are all going to be of this form. An orthogonal polygon with holes is depicted in Figure 2(a).

Given two points p and q in A we say that p is *visible* from q if the line segment joining p and q is totally contained in A, and $||p-q|| \leq R$, where R is the maximal sensor range. A set of points $H = \{h_1, \dots, h_k\} \subset A$ *guards* A if for all $p \in A$ there exists $h_i \in H$ such that p is visible from h_i.

In order to define our minimum time coverage problem we make the following definitions. A *path* P is an ordered set of points $P = p_1, \dots, p_n$. A *convex cover* C of A is a set of convex sets $C = \{c_i\}$, such that $A \subseteq \cup_i c_i$. A convex cover C is *visited* by the path $P = \{p_1, \dots, p_n\}$ if $\forall c_i \in C \; \exists p_j \in P: \; p_j \in c_i$. We define a *maximal* convex cover of A to be a convex cover $C = \{c_i\}$ of A, such that[3] $|c_i| \leq R$, and for all i, there is no convex set $s \subseteq A$, $|s| \leq R$ such that $s \supset c_i$.

An example of a convex cover can be found in Figure 2(b). Note that the four leftmost sets are maximal and overlap, while the sets to the right are disjoint and not maximal. Hence the depicted cover is not maximal. Below we will argue that the overlapping in a maximal cover is preferable in minimum time coverage applications.

Given the above definitions we can state the following lemma before we define the main problem addressed.

Lemma 1. *If there exists a convex cover C of A such that the path P visits C, then P guards A*

Proof. Since P visits C, and every set c_i in C is convex, P guards every set c_i. Furthermore, since $A \subseteq \cup_i c_i$, P guards A. ■

[3] Here, $|s| = \sup_{a,b \in s} \text{dist}(a,b)$ denotes the diameter of the set s.

3 Problem Formulation

In this section we first informally state the Minimum Time UGV Surveillance Problem (MTUSP) and then show that it is $\mathcal{NP}$-hard.

Informally, the problem we are studying is the following: Given a set of surveillance UGVs and a user defined area to be covered, find paths such that every point of the area can be seen from a point on a path and such that the time for executing the search in parallel is minimized.

Problem 1 (Minimum Time UGV Surveillance Problem) *Given N vehicles and an area A, find a set of waypoint paths $P = \{P_1, \dots, P_N\}$ that solve the following optimization problem*

$$\begin{aligned} \min_P \max_i & \sum_{k=1}^{n_i-1} ||p_{ik} - p_{i(k+1)}|| \\ s.t. \quad & \cup_i P_i \textit{ guards } A \end{aligned}$$

Here $P_i = \{p_{i1}, \dots, p_{in_i}\}$ and the start and finish depots, denoted by $p_{i1}, p_{in_i}, i \in \mathcal{Z}_N^+ \triangleq \{1, \dots, N\}$ may be given.

An example solution to a MTUSP can be found in Figure 3.

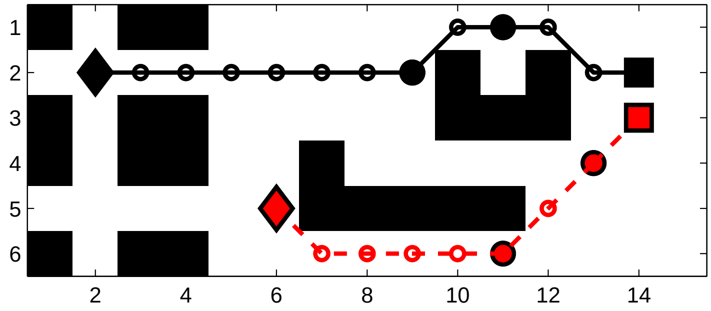


Fig. 3. An approximate solution to the Minimum Time UGV Surveillance Problem (MTUSP) involving two UGVs on the area in Figure 2. Note that all the obstacle free area can be seen from some point on the UGV paths. Details on this and other simulations can be found in Section 5.

Remark 1 (Sensor field of view). In the problem statement above we demand that each point in A is visible from some point in P. This is reasonable in the case of omni-directional sensors. It is however also relevant in the case of cameras mounted on pan-tilt units. In these cases the time right before and after passing p_{ik} must be used to cover the areas visible from p_{ik}. If necessary, the UGVs will have to slow down to facilitate the sensor coverage. A similar argument can be made for the case when the sensor is one or more laser scanners.

Remark 2 (Variations). Throughout this chapter we are focusing on the minimum time coverage problem. However, other closely related problems can also be addressed using the same approach. For instance when surveillance of some particular region has higher priority, or when battery power is a scarce resource and the user wishes to minimize the overall distance traveled by the UGVs. A third option is to make the UGVs avoid high threat areas. All these variations can be incorporated into the solution algorithm presented below by simply varying the considered objective function and edge costs.

We end this section with showing that the problem defined above is $\mathcal{NP}$-hard.

Proposition 1. *Problem 1 (MTUSP) is $\mathcal{NP}$-hard.*

Proof. The proof will built upon a polynomial reduction from an *arbitrary* instance of a well-known $\mathcal{NP}$-hard problem, namely the Euclidean-TSP (ETSP) to a *special* instance of MTUSP[4].

Given an ETSP instance, $(n, [d_{ij}])$, where n is the number of cities to be visited and $[d_{ij}]$ denotes the inter-city distances, we are free to choose the following parts of Problem 1 (MTUSP) such that the achieved optimal solution corresponds to that of the given ETSP.

1. The number of UGVs, N
2. The start and finish depots for all UGVs, $p_1^i, p_{n+1}^i, i \in \mathcal{Z}_N^+$
3. The obstacle configuration
4. The area to be surveyed, A
5. The maximal sensor range, R

Regarding the number of UGVs, $N = 1$ is a natural selection. In order to achieve a tour for this single UGV, we may locate the start and finish depot, p_1^1, p_{n+1}^1 at an arbitrary city cite, as long as they are set equal. Further, an obstacle-free environment is chosen and the area A is taken as the union of isolated points located at the city cites. Finally, we set $R = 0$.

Due to these choices, the area A is fully guarded if and only if the UGV visits all the city locations and hence the optimal solution of this specially designed instance of the MTUSP will coincide with the optimal solution of the given ETSP. This completes the proof. ■

Knowing that MTUSP is $\mathcal{NP}$-hard, we can not hope to solve all problem instances to optimality in reasonable time but must adopt heuristic solution methods. In fact, the comparative study [25] shows that for similar class of problems, finding globally optimal solutions (by commercial software packages like CPLEX) is not a viable approach.

[4] Consult [22–24] to read more about showing $\mathcal{NP}$-hardness through reduction.

4 Proposed Solution

In this section we will propose a solution to the MTUSP described above. The solution encompasses three subproblems, as illustrated in Figure 4. In the first subproblem, the computationally intractable problem of finding the minimum time paths that enable *complete* regional surveillance, is turned into a finite dimensional combinatorial optimization problem. This is achieved by finding a *maximal convex cover* of A, as defined in Section 2. In the second subproblem, the order in which to visit the sets in the cover is determined using Tabu search. The third subproblem, which is called as a subroutine of the second one to evaluate the objective function in the Tabu search, involves a shortest path problem on a graph, constructed from the given visitation order.

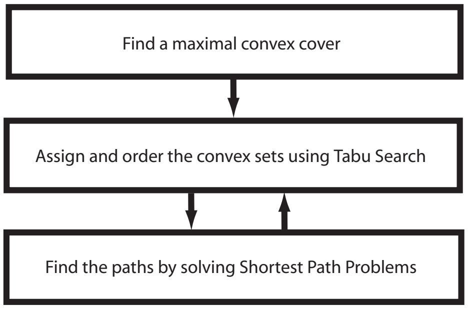

Find a maximal convex cover
Assign and order the convex sets using Tabu Search
Find the paths by solving Shortest Path Problems

Fig. 4. The proposed solution relies on decomposing the problem into three subproblems.

Formally we state the algorithm below.

Algorithm 1 (Proposed solution) *The algorithm consists of the following two steps:*

1. *Create a maximal convex cover* $C = \{c_1, \ldots, c_M\}$ *of* A *in accordance with Algorithm 2.*
2. *Solve the following combined assignment and ordering problem using Tabu search:*

$$\min_\pi F(\pi) = \alpha \max_i f_i(\pi) + (1-\alpha)\Sigma_i f_i(\pi) \tag{1}$$

where π *is a permutation of* $\mathcal{Z}^+_{M+N}$*, representing the assignment/ordering of the* M *convex sets to the* N *vehicles,* $\alpha \in [0\ 1]$*, and* $f_i(\pi)$ *is the optimal path length of UGV* i *given the constraints in* π*. The value of* $f_i(\pi)$ *is found in a sub-routine by using a shortest path formulation to solve the following optimization problem:*

$$\begin{aligned} f_i(\pi) = \min\nolimits_{P_i}\ & \Sigma_k \|p_{ik} - p_{i(k+1)}\| \\ s.t.\ \ & P_i \text{ guards } \cup_{I_i^\pi} c_j \\ & P_i \text{ visits } c_{I_i^\pi(j)} \text{ before } c_{I_i^\pi(j+1)}, j \in \mathcal{Z}^+_{|I_i^\pi|-1} \end{aligned} \tag{2}$$

Here, I_i^π is the index of the sets in C that are assigned to UGV i in the minimization of F. In (1), $\alpha = 1$ corresponds to the minimum time problem and $\alpha = 0$ corresponds to the minimum distance problem. Examples of both these options are found in Figures 10 and 11 in Section 5.

Having stated Algorithm 1 we first note that it does indeed result in a complete covering of the surveillance area A, *i.e.*, it produces a feasible solution to Problem 1. This is clear from the following three observations in conjunction with Lemma 1:

1. A convex cover is created.
2. All sets are assigned to different UGVs in (1).
3. Paths visiting all assigned sets are created in (2).

It is now time to describe and motivate the different subproblems in detail. This will be done in Section 4.1 through 4.3. But first we make the following assumption.

Assumption 1 *Throughout the rest of this chapter we assume that the area A is orthogonal, see Section 2. This assumption is due to the nature of Algorithm 2. Finding a maximal convex cover for a non-orthogonal environment is however not a hard problem and it should be noted that the rest of the solution in Algorithm 1 can handle any general polygon-with-holes type of environment.*

4.1 Finding a maximal convex cover

Since the polygons are all orthogonal, one can see that the maximal convex sets c_i must be rectangles aligned with the polygon. With this fact in mind we can apply the following procedure to find a maximal convex cover.

Algorithm 2 (Maximal convex cover)

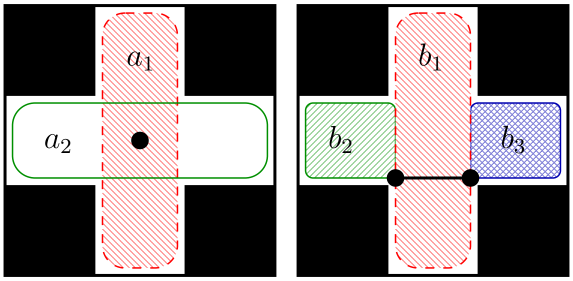
1. *Make a discretization of the area A and construct the corresponding graph representation, $\mathcal{G}(A)$. Since A is orthogonal, a variable sized grid can be created with grid boundaries intersecting all points in the polygon Q and holes $Q_1, \ldots, Q_m$.*
2. *Find a yet uncovered cell, p.*
3. *Start growing a rectangle c_i from p until it is bounded by $|c_i| \leq R$, or the holes on all four sides.*
4. *While uncovered cells exist, goto 2.*

When no more uncovered grid cells can be found the process terminates and A is covered, $A \subseteq \cup_i c_i$. Having described how to find a maximal convex cover in detail, we now discuss a number of related issues.

We first note that decomposing the problem into subproblems, first creating a maximal cover and then finding paths visiting that cover, might remove the optimal solution from the new set of feasible solutions. Since Problem 1 is $\mathcal{NP}$-hard however, our aim is not to solve the problem to optimality, but rather to produce high-quality solutions in reasonable time.

The second thing to notice is that one straight forward solution to the problem would be to first solve a so-called Art Gallery Problem [12] to find a small set of points guarding A, and then solve an ETSP visiting these points. This would however not be efficient, since the points are chosen to be as few as possible, not to permit short vehicle paths. For the same reason, we do not choose a cover with as few sets as possible. That option furthermore happens to be an instance of the optimal *set cover problem* which is one of Karp's 21 $\mathcal{NP}$-complete problems [26].

The benefit of a maximal convex cover is illustrated in Figure 5. If the area A is cross–shaped, as depicted in the figure, then the entire area can be *instantly* surveyed from any point in $a_1 \cap a_2$. Using disjoint orthogonal sets however, the minimum time path for visiting all the orthogonal polygons (b_1, b_2, b_3), and thereby be sure to have surveyed the entire area, is strictly larger than zero.



(a) Maximal convex cover. (b) Rectangular disjoint convex cover.

Fig. 5.

4.2 Assignment and ordering of the convex sets using Tabu search

In this section we describe how we propose to solve the optimization problem in (1), *i.e.*,

$$\min_{\pi} F(\pi) = \alpha \max_{i} f_i(\pi) + (1 - \alpha) \Sigma_i f_i(\pi).$$

Above, π is a permutation representing the assignment/ordering of the sets c_i to the UGVs, and $f_i(\pi)$ is evaluated by solving another optimization problem, as explained in Section 4.3 below.

In order to solve the assignment and ordering problems simultaneously we first give the sets and UGVs id-numbers. Assign the id numbers $1, \ldots, M$ to the convex sets $c_i, i \in \mathcal{Z}_M^+$. Let furthermore the N vehicles have id numbers $M + 1, \ldots, M + N$. The search space for the Tabu search then consists of all permutations of the id numbers, *i.e.*, $\mathcal{Z}_{M+N}^+$. The interpretation of a sequence of id numbers is then best explained by means of the following example: Let

$M = 14$, $N = 3$, and the final sequence be

$$\pi = (\underline{15},\ 1,\ 4,\ \underline{17},\ 10,\ 14,\ 9,\ 3,\ 8,\ \underline{16},\ 2,\ 13,\ 12,\ 7,\ 6,\ 5).$$

This corresponds to the following assignments:

Set with id numbers	assigned to UGV with id number
1 4	15
2 13 12 7 6 5	16
10 14 9 3 8	17

The details of the implementations are, apart from the evaluation of $f_i(\pi)$, identical to those presented in [27]. Hence we refer the interested reader to that paper for a detailed description. We just note that the neighborhood search is performed by pairwise interchanging components in π and the Tabu condition corresponds to requiring a minimum number of iterations before switching a particular pair again.

We now turn to see how the function $f_i(\pi)$ is evaluated in each Tabu step, and how the individual UGV paths are found.

4.3 Path planning and functional evaluation by solving shortest path problems

In the Tabu step above, each UGV is assigned a number of sets c_i and an order of visitation. The problem is now to decide what part of each set to pass through, in order to make the resulting UGV path as short as possible while visiting the assigned sets in the correct order. Formally, we need to solve the optimization problem in (2) for a given assignment and ordering π, *i.e.*,

$$\begin{aligned} f_i(\pi) = \min\nolimits_{P_i} \ & \Sigma_k ||p_{ik} - p_{i(k+1)}|| \\ \text{s.t.} \ & P_i \text{ guards } \cup_{I_i^\pi} c_j \\ & P_i \text{ visits } c_{I_i^\pi(j)} \text{ before } c_{I_i^\pi(j+1)}, j \in \mathcal{Z}^+_{|I_i^\pi|-1} \end{aligned}$$

Given a pair of starting and finishing positions for each vehicle[5], we construct a particular graph for each vehicle. This graph, which is termed a *Route Graph*, has the starting and finishing positions as its first and last node. As depicted in Figure 6, the intermediate nodes are extracted from the ordering π and correspond to the nodes of $\mathcal{G}(A)$ inside the convex sets $c_j,\ j \in I_i^\pi$. To obtain the edge costs for the Route Graph, an "all pairs shortest path problem" is solved in the graph representation of A, $\mathcal{G}(A)$. This can be done by running Dijkstra's algorithm, [22], once with each node of $\mathcal{G}(A)$ as source node.

We illustrate the Route Graph with the following example. Assume as in the table above that the UGV with id number 15 starts from some point p_1 and is assigned to visit first c_1 and then c_4 on its way to p_2. These sets and positions can be found in Figure 7.

[5] For applications indifferent to finishing point, it is possible to let the optimization routine choose it freely.

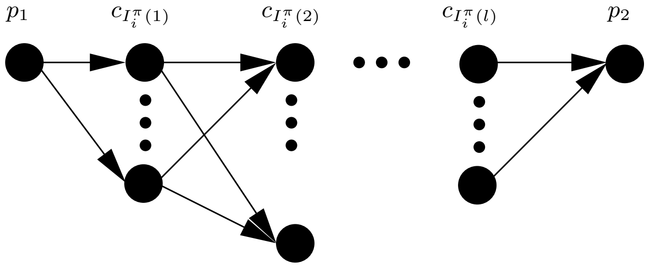


Fig. 6. A graph representation of the route of one UGV.

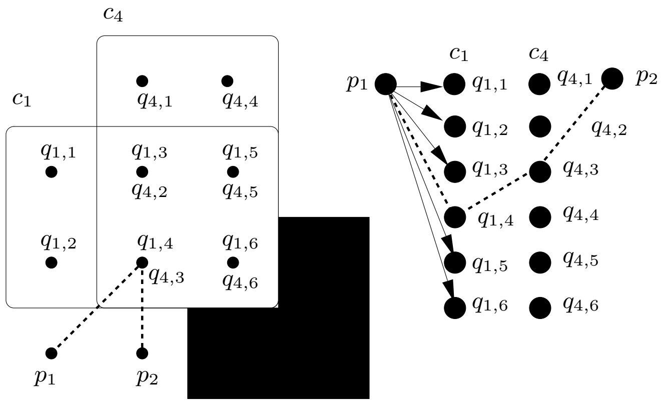


Fig. 7. Example scenario with corresponding Route Graph and optimal path (dashed). Note how the fourth node in c_1, $q_{1,4}$, coincides with the third one in c_4, $q_{4,3}$, hence the cost of the edge between them equals zero in the Route Graph.

The shortest path starting from p_1, visiting at least one node in c_1 and then visiting at least one node in c_4 and finally ending up at p_2, is plotted in the figure as well as in the Route Graph. As can be seen, the fact that c_1 and c_4 overlap makes $q_{1,4}$ coincide with $q_{4,3}$, enabling a very short path.

The evaluation of $f_i(\pi)$ corresponds to solving a shortest path problem in the Route Graph; a task for which polynomial time algorithms such as Dijkstra or A* exist. Note that the solution of this optimization problem yields both the UGV path P_i and its length $f_i(\pi)$.

5 Simulations

In this section, a small selection of examples are presented. The objective is to highlight some of the key characteristics of the proposed solution method. Throughout this section, the search area, A, is chosen to be all of the obstacle free space. Implemented in Matlab, the area representation is normally taken as a random matrix with obstacle density $\rho \in \{0.3 \quad 0.7\}$. It is assumed that the obstacles have been enlarged with the diameter of the vehicle so that paths passing between two obstacles do not imply collision. Furthermore, the initial position of the vehicles are marked with a square (■), while the final positions are marked with a diamond (◆). These two, together with the filled larger circles represent the surveillance points for guarding A.

The first two simulations, found in Figures 3 and 8, illustrate the cooperative nature of the MTUSP. The final positions of the vehicles are here *free* variables to be chosen by the optimization routine. As can be seen, this extra degree of freedom is used constructively so that the vehicles survey the horizontally and vertically aligned "streets" in a cooperative manner with the common objective of minimizing the search time, *i.e.*, we have chosen $\alpha = 1$ in (1). These simulations are also a testimony of the advantage of using a highly overlapping cover.

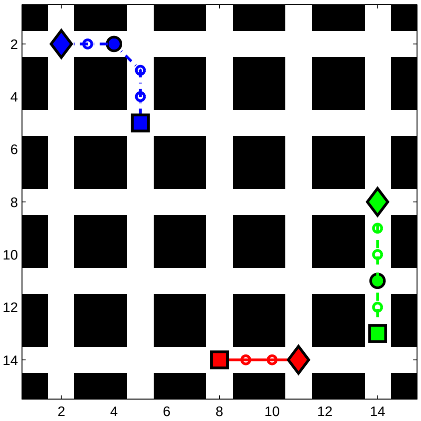


Fig. 8. The Manhattan grid is surveyed cooperatively in minimum time. The starting points of the UGVs are marked with ■.

Figure 9 illustrates a possible drawback of choosing the objective function as pure minimum time. Here, the route of the vehicle to the left, (dash-dotted), is unnecessarily long since a complete coverage would also have been achieved if the vehicle did not move at all. However, the minimum time objective has no way of distinguishing between these two solutions and regards them as equally good since the time for executing them in parallel is indeed equal.
Figures 10 and 11 further illuminate the interplay between the choice of the objective function and the obtained solutions. In Figures 10(a) and 11(a), the solutions are once again obtained by minimizing the total search time. It can be noted that these solutions distribute the work load quite evenly over the vehicle fleet. In Figures 10(b) and 11(b) however, the objective has been set to minimize the total distance traveled by the vehicles, *i.e.*, $\alpha = 0$ in (1). Since this option

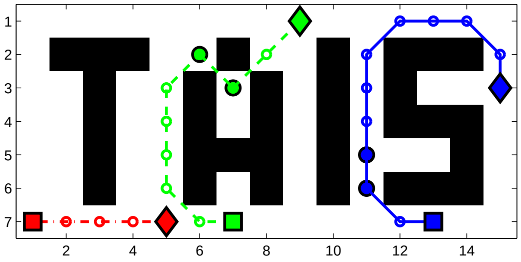
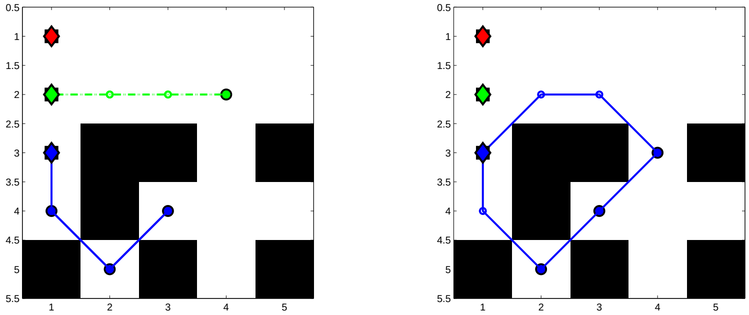


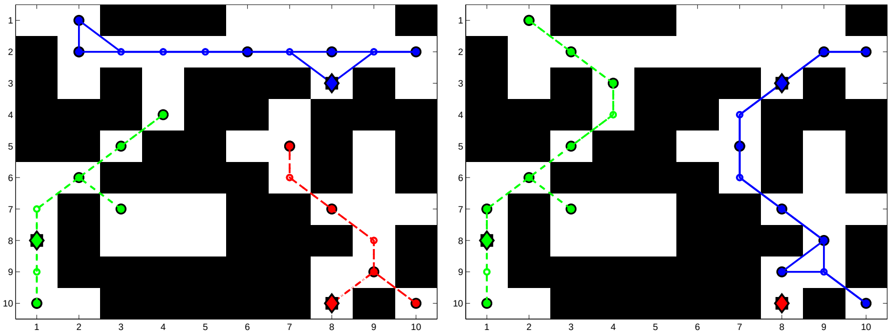
Fig. 9. Minimum time surveillance. Note how the UGVs collectively guard A.



(a) Minimum time objective.

(b) Minimum distance objective.

Fig. 10.



(a) Minimum time objective.

(b) Minimum distance objective.

Fig. 11.

does not take into consideration the division of the work load between the different vehicles, the resulting solutions often do not utilize some of the vehicles at all. This may be of interest when *e.g.*, battery power must be saved, or when unemployed vehicles can be used to perform other tasks.

6 Concluding Remarks

The Minimum Time UGV Surveillance Problem (MTUSP), where it is *occlusion*, and not sensor range, that is the main limitation to the sensors' field of view, is at the focal point of this chapter. We initially show that this problem is in fact $\mathcal{NP}$-hard, hence we cannot hope to solve all instances to optimality in reasonable time. We then proceed by proposing a decomposed solution method that encompasses finding a maximal convex cover, performing Tabu search on the assignment and ordering of the convex cover and finally, solving shortest path problems in the so called Route Graphs. The simulations demonstrate the advantage of using a maximal and highly overlapping convex cover, the cooperative nature of the MTUSP and the interplay between minimum time and minimum distance solutions.

Future research involves various interesting extensions of the current problem formulation, for instance, imposing path-wise constraints that require the induced information graph to be kept recurrently connected.

References

1. E. Acar, H. Choset, Y. Zhang, and M. Schervish, "Path Planning for Robotic Demining: Robust Sensor-based Coverage of Unstructured Environments and Probabilistic Methods." *International Journal of Robotics Research*, vol. 22, no. 7, pp. 441–466, 2003.
2. Y. Guo, L. Parker, and R. Madhavan, "Towards Collaborative Robots for Infrastructure Security Applications," *Proceedings of The 2004 International Symposium on Collaborative Technologies and Systems*, pp. 235–240, 2004.
3. H. Choset, "Coverage for Robotics–A Survey of Recent Results," *Annals of Mathematics and Artificial Intelligence*, vol. 31, no. 1, pp. 113–126, 2001.
4. N. Hazon and G. Kaminka, "Redundancy, Efficiency and Robustness in Multi-Robot Coverage," *Robotics and Automation, 2005. Proceedings of the 2005 IEEE International Conference on*, pp. 747–753, 2005.
5. W. Huang, "Optimal Line-Sweep-Based Decompositions for Coverage Algorithms," *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, vol. 1, 2001.
6. S. Ge and C. Fua, "Complete Multi-Robot Coverage of Unknown Environments with Minimum Repeated Coverage," *Proceedings of the IEEE International Conference on Robotics and Automation*, 2005.
7. D. Carroll, H. Everett, G. Gilbreath, and K. Mullens, "Extending Mobile Security Robots to Force Protection Missions," *AUVSI Unmanned Systems*, pp. 9–11, 2002.
8. V. Ablavsky, D. Stouch, and M. Snorrason, "Search Path Optimization for UAVs Using Stochastic Sampling With Abstract Pattern Descriptors," *Proceedings of the AIAA Guidance Navigation and Control Conference, Austin, TX, August*, 2003.
9. D. M. Panton and A. W. Elbers, "Mission planning for synthetic aperture radar surveillance," *Interfaces*, vol. 29, no. 2, pp. 73–88, 1999.

10. M. John, D. Panton, and K. White, “Mission planning for regional surveillance,” *Annals of Operations Research*, vol. 108, pp. 157–173, 2001.
11. J. Hespanha, H. Kim, and S. Sastry, “Multiple-Agent Probabilistic Pursuit-Evasion Games,” *Decision and Control, 1999. Proceedings of the 38th IEEE Conference on, Phoenix, AZ*, vol. 3, pp. 2432–2437, 1999.
12. I. Bjorling-Sachs and D. Souvaine, “A Tight Bound for Guarding Polygons With Holes,” Report LCSR-TR-165, Lab. Comput. Sci. Res., Rutgers Univ., New Brunswick, NJ, Tech. Rep., 1991.
13. F. Hoffmann, M. Kaufmann, and K. Kriegel, “The art gallery theorem for polygons with holes,” in *32nd Annual Symposium on Foundations of Computer Science (San Juan, PR, 1991).* Los Alamitos, CA: IEEE Comput. Soc. Press, 1991, pp. 39–48.
14. J. Urrutia, “Art gallery and illumination problems,” in *Handbook of computational geometry*, J.-R. Sack and J. Urrutia, Eds. North-Holland Publishing Co., 2000, pp. 973–1027.
15. A. Ganguli, J. Cortes, and F. Bullo, “Distributed Deployment of Asynchronous Guards in Art Galleries,” *Proceedings of the 2006 American Control Conference*, 2006.
16. B. Gerkey, S. Thrun, and G. Gordon, “Visibility-Based Pursuit-Evasion with Limited Field of View,” *The International Journal of Robotics Research*, vol. 25, no. 4, pp. 299–315, 2006.
17. A. Efrat, L. Guibas, S. Har-Peled, D. Lin, J. Mitchell, and T. Murali, “Sweeping Simple Polygons With a Chain of Guards,” *Proceedings of the 11th ACM-SIAM Symposium on Discrete Algorithms*, pp. 927–936, 2000.
18. B. Yamauchi, “Frontier-Based Exploration Using Multiple Robots,” *Proceedings of the second international conference on Autonomous agents*, pp. 47–53, 1998.
19. W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun, “Collaborative Multi-Robot Exploration,” *IEEE International Conference on Robotics and Automation (ICRA)*, vol. 1, 2000.
20. J. Cortes, S. Martinez, T. Karatas, and F. Bullo, “Coverage Control for Mobile Sensing Networks,” *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp. 243–255, 2004.
21. Skoglar, Nygards, and Ulvklo, “Concurrent Path and Sensor Planning for a UAV - Towards an Information Based Approach Incorporating Models of Environment and Sensor,” *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2006*, vol. 1, 2006.
22. C. H. Papadimitriou and K. Steiglitz, *Combinatorial optimization: algorithms and complexity.* Mineola, NY: Dover Publications Inc., 1998.
23. R. Horst, P. Pardalos, and N. Thoai, *Introduction to Global Optimization.* Kluwer Academic Pub, 2000.
24. M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness.* WH Freeman & Co. New York, NY, USA, 1979.
25. J. Thunberg, D. A. Anisi, and P. Ögren, “A comparative study of task assignment and path planning methods for multi-UGV missions,” in *Optimization and Cooperative Control Strategies*, M. Hirsch, P. Pardalos, R. Murphey, and D. Grundel, Eds. Springer, Lecture Notes in Control and Information Sciences, 2008.
26. R. M. Karp, “Reducibility among combinatorial problems,” in *Complexity of computer computations (Proc. Sympos., IBM Thomas J. Watson Res. Center, Yorktown Heights, N.Y., 1972).* New York: Plenum, 1972, pp. 85–103.
27. P. Ögren, S.-L. Wirkander, A. Stefansson, and J. Pelo, “Formulation and Solution of the UAV Paparazzi Problem,” in *AIAA Guidance, Navigation, and Control Conference and Exhibit; Keystone, CO, USA*, 21–24 August 2006.