Multi-robot target detection and tracking: taxonomy and survey — **Source link** ⬈

Cyril Robin, Simon Lacroix

Institutions: Centre national de la recherche scientifique

Published on: 01 Apr 2016 - Autonomous Robots (Springer US)

Topics: Work related

Related papers:

- Search and pursuit-evasion in mobile robotics

- A Survey and Analysis of Multi-Robot Coordination

- Multi-target visual tracking with aerial robots

- Coverage control for mobile sensing networks

- Cooperative Robots to Observe Moving Targets: Review.

Multi-robot Target Detection and Tracking: Taxonomy and Survey

Cyril Robin, Simon Lacroix

HAL Id: hal-01183372
https://hal.archives-ouvertes.fr/hal-01183372

Submitted on 7 Aug 2015

Multi-robot Target Detection and Tracking: Taxonomy and Survey

Cyril Robin[*][†] and Simon Lacroix[*][‡]

[*]CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France
[†]Univ de Toulouse, INSA, LAAS, F-31400 Toulouse, France
[‡]Univ de Toulouse, LAAS, F-31400 Toulouse, France

Abstract—Target detection and tracking encompasses a variety of decisional problems such as coverage, surveillance, search, patrolling, observing and pursuit-evasion along with others. These problems are studied by several communities, that tackle them using diverse formulations, hypotheses and approaches. This variety and the fact that target related robotics problems are pertinent for a large spectrum of applications has motivated a large amount of contributions, which have mostly been surveyed according to one or another viewpoint. In this article, our objective is to go beyond the frontiers of specific communities and specific problems, and to enlarge the scope of prior surveys. We define classes of missions and problems, and relate the results from various communities according to a unifying taxonomy. We review various work related to each class of problems identified in the taxonomy, highlighting the different approaches, models and results. Finally, we propose a transverse synthesis which analyses the approaches, models and lacks that are recurrent through all the tackled problems, and isolate the current main research directions.

Keywords—*axonomy Target Tracking Target Detection Pursuit-Evasion Multi-robotaxonomy Target Tracking Target Detection Pursuit-Evasion Multi-robotT*

I. MOTIVATIONS

Detecting, localizing or following targets is at the core of numerous robotic applications, in both adversarial and cooperative contexts. Much work has been devoted in various research communities to such problems and the related decisional issues, which are often referred to as "pursuit-evasion" problems in the literature. This very evocative term actually encompasses a variety of scenarios that pertain either to mono- or multi-robot contexts, considering either a single or multiple targets, and whose objective is either to detect, to capture or to track them. On the other hand, other similar problems are named differently and make use of specific vocabulary, *e.g.* surveillance, search or tracking. This is partly explained by the different application contexts considered (industrial, civilian or military), and by the fact that different communities tackled them with different standpoints (*e.g.* sensor data processing, symbolic or geometric task planning, task allocation, game theory, *etc.*)

The variety of target related robotics problems and proposed approaches has motivated a vast amount of contributions, and several surveys focused on specific problems are available (Urrutia, 2000; Alspach, 2006; Chung et al, 2011; Miao, 2010; Portugal and Rocha, 2011; Nahin, 2012; Stone, 2007). In this

article, our objective is to go beyond the frontiers of specific communities and specific problems, and to enlarge the scope of prior surveys. Target detection and target tracking, the two broad classes of scenarios related to targets, have *a priori* little to do one with another (and similarly the approaches to solve them), and are often executed in sequence. But in actual applications, these scenarios must be achieved by the same robots, and must even be sometimes tackled simultaneously in multi-robot multi-target scenarios: hence we believe it is relevant to analyse them together.

This article first introduces a coarse taxonomy and the associated vocabulary of the various robotics missions related to targets, so as to explicit how the researches made in different communities relate. It then reviews the various work related to the problems defined by the proposed taxonomy, identifies and relates the main approaches, and finally highlights open areas of research.

Section II introduces the taxonomy, which aims at defining a scope for the analysis of the various problems and approaches related to target management. The taxonomy is driven by the problem (or mission) definitions, and not by the chosen solutions. It is synthetic, and does not distinguish every detail in the problem formulations, but rather gathers them into a global framework, defining large classes of problems and highlighting the main differences. Sections III and IV present the main contributions to the two major classes of problems related to target management, namely target detection and target tracking. For each specific problem of these classes, the main contributions to the state of the art are reported and grouped according to the nature of the developed solutions. Section V then synthesizes the main approaches and associated models, analyses the recurring ones which are common and transverse to the problems, and assesses their maturity in terms of validation. It finally identifies the main open research directions. A conclusion closes the paper.

II. TAXONOMY

Before discussing target detection and target tracking problems, we must first refine the notion of target. In our taxonomy and the following survey, we consider that a target is assimilable to a coherent physical object, whose shape may vary in time but that is somehow "solid" and that can be localised as a point in the environment. This encompasses humans, animals, cars, planes, boats, *etc.*, and also other objects like punctual starting fires or radio beacons. Although we sometimes refer

to work that consider larger fires or chemical spills, this is not the focus of our survey: spatially spread phenomena (like a cloud or a plume) is not considered as a target in this article. This definition excludes *de facto* adaptive sampling problems[1], but help to contain the still very large scope of our survey.

Hence, typical robotics target related scenarios are automated surveillance of secured areas, frontier patrolling, secured area clearing, target tracking or chasing, *inter alia*. In all these scenarios the environment is mostly known, and exploring the environment is not considered. The targets may either be mobile or fixed, but we focus in this survey on mobile targets, which are more challenging.

The taxonomy pertains to the decisional aspects of the problems and is summarized in Figure 1: it is organized as a tree, in which each branching is defined by a specific criterion. Each leaf refers to a class of problems, including possible variations in the formulation or assumptions. It defines coherent notations and definitions of the problems used throughout the article. Note that even though we try to comply with widely accepted vocabulary, there may be conflicts with definitions used by some authors. The names and definitions of the various problems are indeed not standardised, especially when considered in different communities: the same word may refer to different problems in the literature, and so we stick to the taxonomy vocabulary throughout the paper.

The first branching criterion of the taxonomy relates to prior knowledge on the target position, and yields the two main classes of problems, that often occur in sequence: detecting targets on the one hand, and tracking detected targets on the other hand.

A. Target Detection

Target detection problems consist in finding (*detecting*) a target in a given environment. They may concern one or several targets and may be tackled with one or multiple sensors, either by actively sweeping the environment with mobile sensors, or by monitoring signals emitted from fixed static sensors.

We refer to this later class of problems as *static surveillance*: it mainly involves sensor positioning strategies, which often come to partition the environment and accordingly distribute sensors within this environment.

When mobile sensors are exploited, the problem is strongly related to path planning, and we refer to it as *mobile search*. Such problems can be addressed either locally or globally. Depending on the models, the assumptions and the approaches, some authors try to provide worst-case guarantees for the performance (*capture*), whereas some provide probabilistic guarantees (*probabilistic search*) and others do not provide any guarantee at all (*hunting*). The *search* may have a cyclic aspect (*patrolling*), although this presents no interest for *capture* or *hunting*. Note that for each of these problems, a variant that consists in "surrounding" the target in order to prevent its evasion is sometimes considered, instead of merely watching it, or catching it.

---

[1]Adaptive sampling or information gathering processes consist in assessing the extend and/or distribution of spatially spread phenomena – see (Hollinger and Sukhatme, 2013) for instance.

Static surveillance: In *static surveillance*, the environment is necessary known and the objective is to optimally position a set of fixed sensors. The traditional form of *static surveillance* is the famous *Art Gallery* problem which has been well studied and for which numerous results have been obtained – see Section III-A. Variants of *static surveillance* include mobile sensors, but the proposed solutions always focus on the sensor placement aspect (*where* to set the sensors?), and not on path planning (*how* to reach the selected positions?).

Capture: In the *capture* problem, optimality and completeness are essential characteristics. The goal is to clear a given known area while providing a worst-case guarantee, meaning that if a target is inside the considered area, it will be found, no matter what. There is no prior knowledge or assumption on the target location, targets may even have "super abilities" (like infinite speed), and pursuers try to surround them. *Capture* is often referred to as a *pursuit-evasion*, but also as *search and secure*, or as the *cops and robbers* game, mostly when the solution rely on *graph clearing* (Chung et al, 2011).

Work on this subject usually has strong mathematical foundations, and are developed following two main approaches (Alspach, 2006; Chung et al, 2011): stating the problem as a *graph clearing* one, or as a purely geometric one, within 2D polygonal environments. Often, the objective is to assess the minimal number of robots required to provide a worst-case guarantee. Section III-B further discusses approaches to *capture*.

Probabilistic Search: The main difference between *capture* and *probabilistic search* is the absence of worst-case guarantee in the latter, in which *probabilities of detection* are assessed (Stone, 2007). The reason is mainly a lack of resources (robots or time) to tackle the worst-case problem, but it can also be a compromise between efficiency and the probability of occurrence of particularly difficult situations. *Probabilistic Search* exploits probability distributions over the model of the environment (of the target presence, of the target visibility, *etc.*) Most authors try to provide bounds on the probability of detecting/catching the target. The target model may either be adversarial or not, the latter being usually easier to deal with because of its lower algorithmic complexity. The non-adversarial target model is widely used in *search and rescue* scenarios, for which emergency and time constraints usually prevent performing an exhaustive search and impose priorities – which is well handled by probabilistic models. An extended description of the problem and associated algorithms is given in Section III-C.

Patrolling: When the mobile search is cyclic, it is denoted by *patrolling*. *Patrolling* may be seen as a cyclic version of *probabilistic search*, as it involves analyses of statistical performance over time, and especially the time elapsed between two visits to the same point. It is a rather recent area of research, whose interest has risen over the last decade (Portugal and Rocha, 2011). Related work are discussed in Section III-D.

Hunting: There are finally some cases where no guarantee at all is provided for detection or capture of the target, which we refer to as *hunting*. The absence of guarantee comes from the lack of resources (robots, time) or information – in the absence of which no useful probability models for the target location
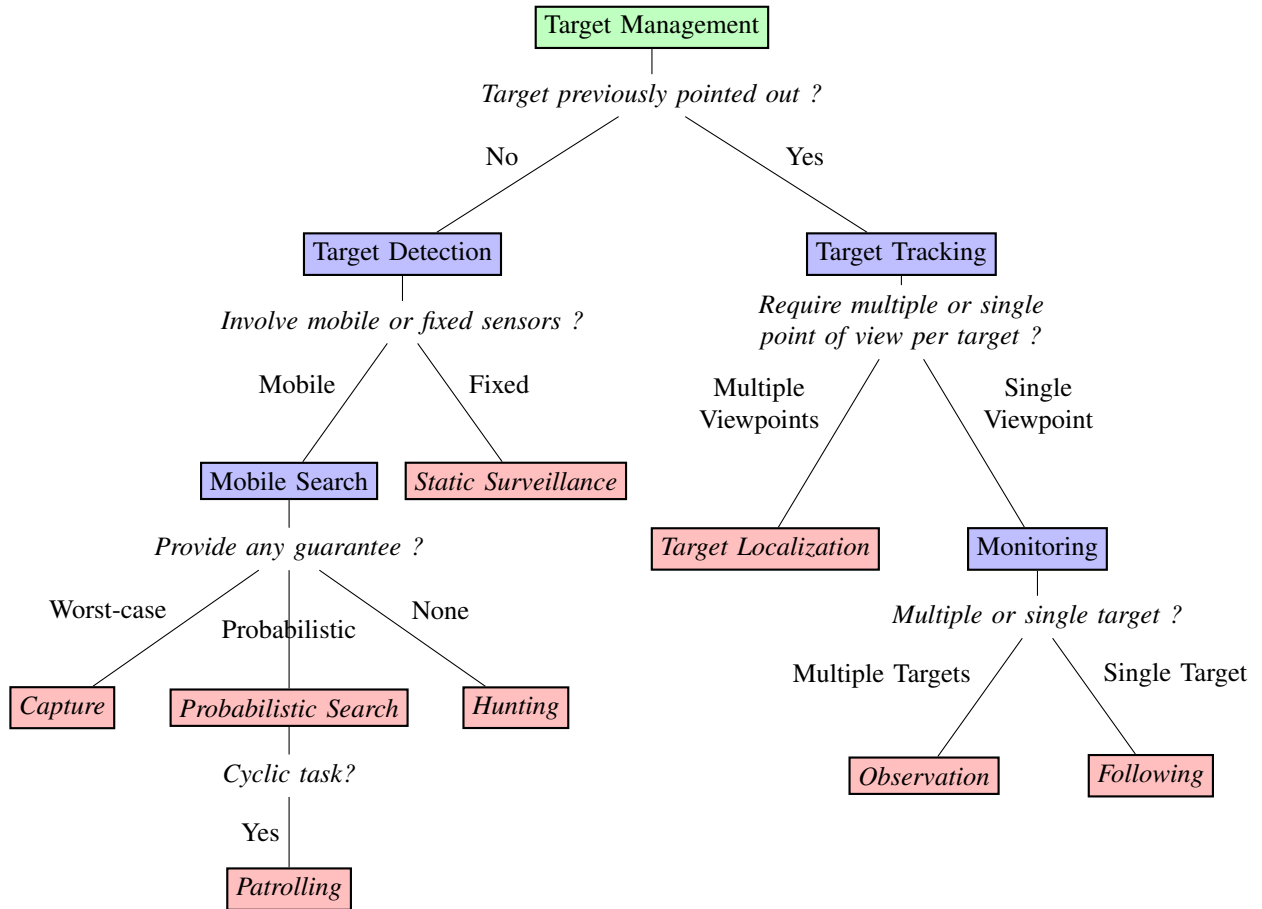
Fig. 1. Proposed taxonomy of the target management problems. Branchings correspond to criteria, here denoted by questions, each read leaf correspond to a specific *problem* analysed in the following sections.

can be exploited, for instance. *Hunting* is often considered within a multi-robot context; related work in this area are presented in Section III-E.

B. Target Tracking

The second major class of problems, *target tracking*, corresponds to the tasks that arise when one or several targets have been detected or assigned – often following the success of target detection tasks. Note that the target tracking task itself is an estimation and control problem, not a planning problem and is thus not addressed here. In our survey, coping with a target may imply keeping it in sight, to provide information on it (mainly to localise it over time, but identifying it can also be an objective), or to catch it. In all cases tracker robots need to stay "close" to the targets, the required distance being zero when it comes to catch the targets. Coping with a single target may require one or more robots, depending on the context. It is for instance preferable to have multiple vantage points on each target to refine their locations. We refer to this latter class of problems as *target localization* problems. We also distinguish one vs. one problems (*following*) from the multi-robot multi-target problems (*observation*).

Target localization: In *target localization* problems, the goal is to track a target with several robots in order to improve knowledge about the target, in particular the precision of its estimated position. It is most often a multi-robot problem, in which case solutions involve selecting different points of view to maximize the information gain. Of course, data fusion, cooperation, communication and multi-robot localization are issues to be considered. Several targets may be involved, and several observation points of view on each target are often required. Multi-robot *target localization* has naturally gained interest with the development of research on multi-robot systems (Section IV-A).

Following: The class of *following* problems is the traditional form of pursuit-evasion. Alexander presents a brief history of related work in (Alexander et al, 2009), and Nahin gives a more complete history and a more extensive survey in (Nahin, 2012). Early work pertained to naval conflict scenarios, and the traditional pursuit problem is also known as *Lion and Man*.

In the original version, a single purser (the Lion) is chasing a single evader (the Man) with the same speed. Many different versions of the problem have been defined, with different speeds, environment models, visibility conditions, *etc. Follow-*

ing embraces all these variations, which are tracking problems involving only one pursuer and one evader (Section IV-B).

Observation: The *observation* problem is stated as follows: given several robots and several (moving) targets, how to control the robots in order to simultaneously observe all the targets, and if not possible, how to minimize the time during which any target is not observed by at least one of the robots. This problem has been rigorously specified by Parker in 1997 (Parker and Emmons, 1997), who refers to it as the CMOMMT problem (*Cooperative Multi-robot Observation of Multiple Moving Targets*). One of the main challenges is to correctly allocate targets to robots and to decide how and when the observers should trade targets (see Section IV-C).

III. TARGET DETECTION

This section describes more extensively the left side of the taxonomy, and presents the main associated work. We do not aim at being exhaustive (several surveys on the problems addressed are available (Urrutia, 2000; Chung et al, 2011; Miao, 2010; Portugal and Rocha, 2011; Stone, 2007)), but rather try to emphasize the main work and also some interesting uncommon approaches. We also highlight the recurring limitations in the work on the subject, although this will mainly be developed and discussed in Section V.

A. Static Surveillance – Art Gallery

We define *static surveillance* as optimally positioning a set of fixed sensors in a known bounded environment in order to cover the whole area while using a minimal number of sensors. The literature also refers to this as the *art gallery* problem and sometimes the *coverage* problem. However, this latter term is tricky as it is also widely used to refer to the *coverage path planning* problem (Choset, 2001) that we do not address here, which goal is to determine a path to cover all points in a free space without any notion of target. We also present some variations of *static surveillance* which differ from the original *art gallery* statement.

Art Gallery: The famous *art gallery* problem was initially proposed in 1973 by Victor Klee as a response to Vasek Chvátal who requested an interesting geometric problem (O'rourke, 1987). The problem was to determine the minimum number of guards necessary (or sufficient) to cover the interior of a n-wall polygonal art gallery room (see Figure 2). Chvátal first proved that $\lfloor n/3 \rfloor$ guards are occasionally necessary and always sufficient to guard an art gallery represented by a simple polygon with n vertices. Since then many variations and results were published, including various shapes of the environment and various constraints on the guards (in positioning and sensing abilities).

The problem has been proven to be NP-hard in its general form (O'rourke, 1987). Therefore if nowadays one has enough results to know the number of guard required, finding their proper locations is still complex. The main results and the history related to the *art gallery* problem and its numerous variations have been gathered and discussed in the survey of Urrutia (Urrutia, 2000). This survey also presents the problem of mobile guards and develops the relation with the original *art*
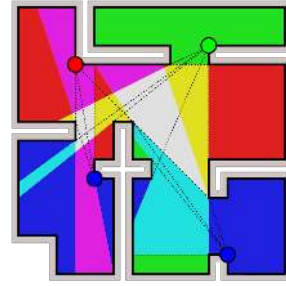

Fig. 2. An instance of the *art gallery* problem, with four guards – from the Wikipedia Art Gallery problem article.

gallery problem. Hence the results on mobile robots gathered by Urrutia mainly tackle the minimal number of required guards rather than to the proper location or path of these guards. For the latter, the reader will find further comments on mobile guards in Section III-B and Section IV-B.

The *art gallery* problem has a great impact on many engineering applications, and several authors try to effectively solve the *static surveillance* problem, *i.e.* to give a positioning strategy for the guards. For instance, Gonzales *et al.* approximate the *art gallery* problem to find good locations for visual sensing (González-Baños and Latombe, 2001). They use a sampling based approximation to transform the original problem into a *set cover* problem[2]. They justify their approach by the complexity of the problem and by the notion of the "elasticity" of the optimum: most of the time the optimal positions may be slightly moved without any noticeable change in the solution quality. And when it is not the case, the given solutions are not desirable for engineering applications because they lack robustness. The reader may refer to (Ghosh, 2009) for more work on approximation algorithms that transform *art gallery* problems into set cover problems.

Variations on static surveillance: The historical version of *static surveillance* is the *art gallery* problem, but more recent work has tackled interesting variations. While some papers reviewed in this section use mobile sensors, the approaches are mostly static, either because they mainly use static sensors and mobile sensors only help these static sensors, or because the solution consists in the definition of several static positions. In other words, these approaches are about *sensor positioning* and not *sensor path planning*.

Rybski's work is not based on classical *art gallery* results, but rather uses an interesting combination of scouts and rangers (Rybski et al, 2000). Both are mobile sensors, but the ranger is a larger, "heavy-duty" robotic platform which can deploy and supervise several small and cheaper scout robots. Once deployed, the goal of scouts is to find dark areas to hide and take position to detect any incoming intruder. The dark area are detected locally, and as lighting conditions change, the scouts may adapt their position, but basically, the final state consists

---

[2]The *Set Cover Problem (SCP)* is a classical question in combinatorics: given a set of elements (called the universe) and a set S of n sets whose union equals the universe, the set cover problem is to identify the smallest subset of S whose union equals the universe.

of static positions for the scouts. The ranger may patrol to gather data between its scouts.

Kloder and Hutchinson tackle and formally define the problem of *Barrier Coverage* (Kloder and Hutchinson, 2007): it is very similar to *art gallery*, but the goal is not to statically cover areas but rather boundaries (*ie* segments in 2D). The goal is to prevent any undetected intrusion in a predefined region. They address the problem for a 2D polygonally-bounded region and solve it by finding the minimum-length barrier, building a graph of candidate barriers and reducing the problem to a maximum-flow/minimum-cut network problem. We find this approach interesting because if one assumes that intruders cannot pop-up anywhere in the region to cover, *barrier coverage* may require fewer guards than an *art gallery* approach. Nevertheless, this seems less robust to tricky threats like intruder dropping, and is not suitable for area surveillance, for instance to detect fire starts.

Pimenta *et al.* address the problem of *Simultaneous Coverage And Tracking* (SCAT) (Pimenta et al, 2010). They actually use mobile sensors to cover a given region like in the *art gallery* problem in order to detect any intruder. The static guarding positions of the robots are determined by a Voronoi tessellation of the map. The tessellation suits the number of available robots. If any target is detected, it is attributed to one robot according to its position in the tessellation. The target positions influence the guarding positions of the robots by modifying a time-varying density functions on which the tessellation is based. Thus both *static surveillance* and *tracking* (or more precisely the *observation* problem defined in Section IV-C) are simultaneously performed. This approach is uncommon, but highlights the links between the left and the right parts of the taxonomy, and the interest in studying these problems together. The authors presents interesting results from simplified simulations, from Gazebo simulations[3], and from tests on-board real robots. However, every tested environment seems to be obstacle free which is a great limitation, as the presence of obstacles or the consideration of a cluttered environment raise difficult problems.

When using static sensors to monitor an area, there can remain holes in the coverage, especially because one lacks sensors, or because the environment configuration does not allow placing sensors nearby. Moreover one may want to have several points of view on the target, but to guarantee this for any point of the covered region with only static sensors would probably be costly. Shi considers a hybrid wireless network, using both static and mobile sensors (Shi et al, 2010). Several sensors are required to track the target, and static sensor clustering is not always sufficient. Therefore mobile sensors are used to patrol the holes in the static coverage, and add extra points of view when needed.

More recently, Bhattacharya *et al.* focus on more complex environments for which the traditional Euclidian distance is not adequate (Bhattacharya et al, 2013). They give interesting results valuable for Riemanian Manifolds with boundaries. However, if most of their work consider a continuous space, they highlight the need for discretization, which is "an indis-

pensable trade-off" to make any continuous problem computationally feasible. They also present an extension to define an exploration scheme.

Formulation as an optimization problem: We saw that the *art gallery* problem is strongly related to the *Set Cover Problem* (González-Baños and Latombe, 2001). The latter may be consider both as a decision problem and as an optimization problem, and more generally, one can formulate any *static surveillance* problem as an optimization problem. This is particularly suitable as the final output is a static solution (a set of static positions). For instance in (Renzaglia, 2012), Renzaglia formulates *static surveillance* as an optimization problem. He uses *cognitived-based adaptive optimization* to solve the problem. Although optimization techniques are mainly centralized the author also proposed a decentralized version of the algorithm – as well as an extension for the 3D case.

Summary: Static Surveillance is a well studied problem. There have been much work since its initial formulation as the *art gallery* problem, and numerous variations have been studied. The approaches are rather theoretic and formal. The main results indicate the minimal number of guards required, and some give strategies about their positioning. More recently mobile sensors have been introduced as a complement and a backup for the static sensors. Despite the various theoretical results, there is a lack of practical results, and one may regret a lack of realism in the models, for instance with respect to the sensing capacities and the communications. That may deeply affect the efficiency of the algorithms in an operational context.

B. Capture – Clearing

The *capture* problem, as defined in the taxonomy, aims at clearing or securing a given known area, capturing all the targets inside. The target positions are not known by the pursuers, and capturing a target may imply surrounding it with enough searchers (or in some cases, seeing/catching the target is enough). As stated in Section II, the notion of optimality and worst-case guarantee are fundamental in the definition of *capture*.

Early work: In 1978, T. D. Parsons published *Pursuit-Evasion in a Graph* (Parsons, 1978). He imagined "a man is lost and wandering unpredictably in a dark cave", while "a party of searchers who know the structure of the cave is to be sent to find him", and ask for the minimum of searchers required to find the lost man regardless of his behaviour. Our definition of *capture* inherits from this statement: the environment is known, the searchers do not know the target location and worst-case guarantee must be provided (the target motions being unpredictable). The problem is to find the minimum number of pursuers to tackle the mission, rather than defining explicit or efficient pursuing strategies.

Parsons pictures the environment as a graph or a tree, and provides several theorems to determine the minimum number of searchers for a given graph recursively considering the search number of their subgraphs. This number is also known as the *search number* of a graph. Parsons' results mainly focus on trees, which have a structure easier to deal with than generic

---

[3]Gazebo is a realistic robot platform simulator (Koenig and Howard, 2004).

graphs. Many interesting results and relations are given, but no global strategy for the searchers is provided.

Finding the search number of a graph has been proven to be NP-hard on general graphs and even NP-complete for graphs with a maximum vertex of degree of 3 (Megiddo et al, 1988), while LaPaugh proved that "recontamination" does not help (LaPaugh, 1982). The contaminated areas are the areas where the target may be, whereas the cleared areas are free from any target. Recontamination is the fact that a previously cleared area may be re-invaded by a mobile target. One could think that allowing this process may ease the global strategy, but it actually does not. This result is crucial as it gives an important hint in the determination of a global strategy for the searchers.

Graph Clearing – the cops and robbers *game:* Inspired by the early work on discretized environment models, many people have worked on *graph clearing*, which is an instance of *capture* where the environment is depicted as a graph. Variants include edge-search (sometimes referred to as *graph sweeping*), node-search (generally called *graph search*), infinite-speed evaders, *inter alia*. The problem is also known as the *cops and robbers* game (Chung et al, 2011). The reader may also refer to (Alspach, 2006) for a survey.

As explained by Moors in (Moors et al, 2005), one difficulty is to transform the geometrical environment into a graph – which can be solved thanks to algorithms pertaining to the *art gallery* problem or through partitioning processes like the Voronoi diagrams (Kolling and Carpin, 2008). Moors also underlines the interest of using "blockers", *i.e.* to "sacrifice" some searchers to guard borders between cleared and contaminated areas. This is important in a *divide-and-conquer* approach which breaks the complexity of dealing with large areas.

Borie *et al.* analyse specific graph configurations for which polynomial-time algorithms exist to determine the search number of the graph (Borie et al, 2011). For instance, finding the search number is linear for specific graphs like trees, interval graphs or grid graphs. The authors exhibit an interesting table which summarizes the complexity results for various types graphs, considering the problem of finding the "minimum pursuer" (*i.e.* the *search number*), the minimum distance, or the minimum time required. They also present an optimal algorithm for the *search number* problem on trees, and extend their result to unit-width / arbitrary-length and arbitrary-width / unit-length graphs, either providing an optimal polynomial-time algorithm or proving the NP-hardness of the problem for various topologies.

Topological considerations significantly help to reduce the complexity and provide efficient, even optimal, solutions. For instance Daniel *et al.* use series-parallel graphs to develop a heuristic approach that scales well on the graph size (Daniel et al, 2010). They get an $O(nr^6)$ complexity where n is the number of graph nodes and r the number of robots.

When the topology is not specific, *i.e.* does not adhere to a well-known class for which efficient algorithms exist, one can find a way to modify and "improve" it, especially by *blocking* – guarding – edges or vertices. For instance, Hollinger *et al.* use guards to transform a graph into a tree, in order te consider a linear problem instead of the NP-hard general case (Hollinger et al, 2010). More specifically, they use spanning trees, and

the proposed algorithm has anytime properties and provide guarantees for some of its variations. One may also notice that they rely on few communications, the robots coordination being partially implicit. In a similar way, Katsilieris *et al.* highlight the importance of guarding vertices and transform the graph depicting the environment in a tree by placing stationary robots to remove loops (Katsilieris et al, 2010). They also analyse the trade-off between time and the number of robots required to clear the environment.

Usually, the proposed graph-clearing algorithms are depicted as *edge-searching*, where robots slide along edges and guard vertices. However, edge-searching algorithms are difficult to implement in the real world since "guarding a vertex performs two basic functions, namely, the prevention of spread of contamination from and to its neighbours and the detection of all intruders in the vertex" as stated by Kolling and Carpin in (Kolling and Carpin, 2010b). Hence they propose Graph-Clear, where a robot can either block an edge or sweep a vertex whose every edge is blocked. Doing so, the search of a region and the prevention of recontamination from neighbouring regions are separated, and the latter occurs on edges. The authors also prove that the decision process related to *GraphClear* is NP-hard in the general case, but for trees they provide an algorithm quadratic in the number of vertices to compute a valid strategy. Later, they also explicitly study the graph construction from occupancy grid maps, trying to suit the graph to the *GraphClear* algorithm to improve the computable strategies (Kolling and Carpin, 2008).

Additional variations of the *graph clearing* problem may be found either in the knowledge of the agents, or in the underlying environments modelled by the graph. For instance, while most variants model the environment as 2D, Kolling considers 2.5D visibility (Kolling et al, 2010), which is particularly appropriate when considering outdoor environments. His approach is based on sampling the environment to build a graph representation where areas slightly overlap to allow better team cooperation. Previously published algorithms are then used to compute the graph-clearing strategies. Vieira *et al.* study a version in which players (pursuers and evaders) have complete knowledge of other players' states, and have optimal strategies according to this knowledge, computed through a state-based game graph (Vieira et al, 2009). However the transition diagram is computed off-line: only local state transitions are computed on-line. Furthermore, the optimal strategy scales exponentially with the number of players.

For more references related to this graph approach of the *capture* problem, the reader may look at the state of the art section of (Moors et al, 2005; Kolling et al, 2010) or in the related *cops-and-robbers* section of Chung's survey (Chung et al, 2011).

Geometry-based approaches: Besides the abstract formulation of the *capture* problem found in graph-based approaches, other work use a continuous representation of the environment, where geometrical considerations are central. This kind of representation allows a deeper insight into the sensor models (field of view) and the details of the strategy, as it is more low-level than the graph abstraction. The standard environment considered here is also a polygonal 2D environment, with

obstacles both for the agents' movement and sensing. Most of the time, the algorithms focus on only one pursuer, and some extend the resulting strategy to several robots.

The question "how many pursuers are required to clear an area" is strongly related to the *art gallery* problem (see Section III-A) and many results on this subject can be found in Urrutia's survey (Urrutia, 2000). Here we focus on the related motion strategies and control of the robots.

Gerkey *et al.* focus on the impact of a limited *field of view* (FoV), which drives their approach (Gerkey et al, 2006) (see figure 3). They introduce the notion of ϕ-searchers, whose field-of-view angle is bounded by ϕ (the length of the FoV is not bounded). They focus on a single ϕ-searcher, and then extend the approach to several robots in a centralized manner. Durham extends the concept of ϕ-searcher to the (d, ϕ)-searcher by taking into account the length d of the FoV (Durham et al, 2011), and extends the work with multi-robot cooperation, including a decentralized approach. It is worth noticing that the approach does not assume the environment is known, but it assumes that enough robots are available (which actually implicitly implies some knowledge about the environment).
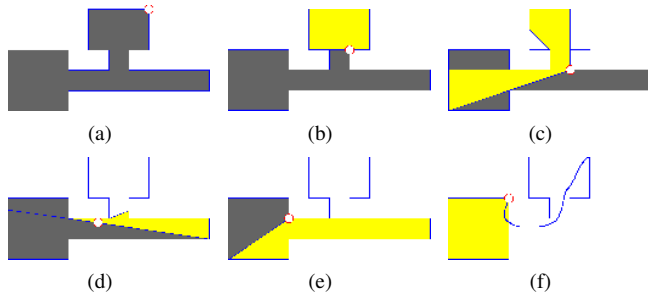

Fig. 3. An instance of environment clearing by a searcher with a ϕ-limited field of view (where $\phi = \pi$) *"In this case the searcher clears the environment by moving backward out of the upper room, down the hall, and into the left room."* – excerpt from (Gerkey et al, 2006).

Note that even though this statement of *capture* is related to *following* (as in Chung's survey (Chung et al, 2011)), there remains a strong difference: here the behaviour is environment centered and the target location is unknown, whereas in *following* the initial target location is known and the pursuer movement is mostly driven by the target one (see Sections II and IV-B).

Decentralized cooperation algorithms: The work on capture presented so far have a drawback: the problem is considered in a centralized manner. As stated by Gerkey in (Gerkey et al, 2006): "planning in the joint configuration space of all searchers is clearly not the best approach, as it is centralized and scales badly as the number of searchers increases". Tanner proposes a partially decentralized approach based on flocking algorithms to control an heterogeneous team where AGVs and AAVs[4] have distinct roles (respectively blockers and sweepers) (Tanner, 2007). On his side, Durham *et al.* use an exploration-like sweeping pattern, based on frontiers, but where robots are constrained when expanding frontiers in order not to leave unwatched frontiers. They also distinguish several dynamic roles, including leaders and followers, which allows to distribute the centralized algorithm and perform decentralized clearing (Durham et al, 2011).

Cooperative decentralized algorithms are well suited to the *capture* problem. For instance, Zhou bases his approach on the *Extended Contract Net Protocol* (CNP) (Zhou et al, 2004). Kalra proposes Hoplites (Kalra et al, 2005), a market-based approach where the robots clear an area by forming a sweeping front-line for which active coordination is required to allocate the free spaces between obstacles.

Static targets: In the very specific cases where one is guaranteed that the targets are static, one can apply another kind of strategies related to the *coverage path planning* problem (Choset, 2001). Such problems aims at determining paths for the robots that make them cover each and every point of a given free space. This kind of strategy is purely environment-oriented and is thus only valid for static targets – otherwise no guarantee can be provided. Nonetheless, coverage path planning implies a systematic coverage of the environment and free space. A classic mono-robot covering strategy is the Boustrophedon path illustrated figure 5a (Choset, 2000)

On-board real robots: As stated in the introduction of this section, various algorithms have been integrated on-board robots. They show interesting performances, which advocates their maturity. For instance, Kalra embeds a decentralized market-based approach on-board Pioneer robots (Kalra et al, 2005), while Vieira runs algorithms on a team of iRobot Create robots, and a full network providing communication and "sensing-at-a-distance" abilities in the whole indoor environment (Vieira et al, 2009). Due to the poor sensors of the robots, Vieira uses wall-following motions, which is sufficient in the considered office environment. Katsilieris uses a large but simple outdoor environment (Katsilieris et al, 2010), whereas Durham's experiment takes place in an *ad hoc* indoor environment, using a previously tested architecture: the Multi-robot Integration Platform along with the Player-Stage framework (Durham et al, 2011).

Katsev *et al.* address the problem in relation to the sensing community (Katsev et al, 2011). They use a wall-following robot, without sufficient sensors that would allow precise mapping or localization. Introducing the notion of *shadow* (where an evader is not detected when behind an obstacle), they compute the global strategy on a map that allows area clearing, given that it is possible with a single robot in the considered environment. Their algorithms are implemented on the Lego NXT robot. The computed plans are worst-case guaranteed, yet the completeness of the algorithm has not been proven.

Summary: Among the numerous papers tackling *capture*, one can distinguish two main approaches: the ones that reason on abstract discretized graphs, and the ones that reason on continuous geometric representations. Graph-based approaches provide numerous complexity results for various configurations defined by on the graph topology. The geometry-based approaches provide less complexity results and guarantees, but allow taking into account more realistic sensor models. Unfor-

---

[4]Autonomous Ground/Aerial Vehicles.

tunately, most of the approaches are still very centralized, but as more and more centralized efficient algorithms are provided, other decentralized approaches and more difficult geometrical configurations (like restricted fields of view or 2.5D visibility) are considered. Finally, some algorithms have been integrated on-board robots, which testifies, at least partially, to their validity.

C. Probabilistic Search

For target detection problems, probabilistic models are obviously useful to handle uncertainties and priorities. The work gathered in this section use a distribution of probability over the world representation in order to represent the possible target locations. A probabilist framework provides some guarantee to detect the targets when the worst cases cannot be handled (*e.g.* because of one lacks robots). Note that this section does not address the uncertainties which come from models and are not part of the problem, *e.g.* when sensors' limitations may lead to missed detection or false positive. This kind of uncertainties indeed leads to considerations that are not problem dependent, and thus are transverse to our taxonomy.

The most common representation of the probabilistic state of the targets is a distribution of probability over a Cartesian grid map (or another discrete space representation, such as tessellation). The evolution of this distribution over time requires models for the target motion. Markovian models are widely used, but their consideration leads to solutions with a high complexity. Static and randomly moving targets models are stationary, in the sense that they do not depend on the target or pursuer history, and hence reduce this complexity.

The probability density function of a target presence is a natural estimate of its localization, and thus several work presented here aim at detecting targets and refining their positions (this is also naturally exploited for the *target localization* problem – see Section IV-A). Even though a prior estimation of the probability distribution is available, the targets positions are not known in the sense explained in Section II: the goal of the mission is indeed to find the targets.

The targets may be adversarial or not. Actually, much work in *probabilistic search* consider a target indifferent to the tracker, mainly because it reduces the complexity; it indeed becomes a one-sided game, as defined by Benkoski in (Benkoski et al, 1991). Such kind of target is also suitable to search and rescue scenarios (the reader may refer to (Benkoski et al, 1991; Chung et al, 2011) for further discussion).

A team of robots naturally offers numerous advantages over a single robot. Although multi-robot approaches raise several issues on the scalability of the algorithms, on the centralization of the systems and on network connectivity, they appear to be the *de facto* norm in the *probabilistic search* context.

Despite the vocabulary used by some authors, *probabilistic search* is not *capture* as defined in our taxonomy: for instance in (Bopardikar et al, 2007) the trackers are not numerous enough to provide worst-case guarantees like in *capture* (or at least there is no intention to deal with the worst cases), thus only bounds on the probability of catching the target are provided. In spite of the methodical sweeping movement

of Bopardikar's pursuers, they cannot prevent the target from escaping, and only provide bounds on the capture probability. Nevertheless, these bounds are important to evaluate the efficiency of the algorithms.

The probabilistic framework also offers two advantages: first it eases the implementation on real robots, naturally handling uncertainties in perception and action (more realistic models). Second it allows prioritizing tasks when time matters and prevents performing the complete task, *e.g.* in *search and rescue* mission where the goal is rather to find as many victims as possible in a very short amount of time, than to find each and every victim, which is most often hardly realistic. One may talk about "probabilistic completeness".

Early work: Early work in *probabilistic search* were driven by the idea that sometimes the map is not given nor known, and thus a graph representation does not suit well. Hespanha *et al.* consider a probabilistic pursuit-evasion game in the sense of "target detection" (Hespanha et al, 1999). More precisely, they consider random walk evaders in a "game against nature", which actually embodies *search and rescue* problems. Their agents use a greedy policy, driven by the maximization of the conditional posterior probability, based on an inaccurate *a priori* grid map. The authors use a probability distribution to embody the possible target locations on this map, which evolves as the robots explore the environment. They handle probabilistic models of the motions and sensors, and both space and time are quantized. The authors also introduce the important notion of *"persistent policy"*, in which every move must always ensure a chance to capture the target in the future (no "make-or-break" move). Although the solution is not optimal, the approach is mathematically grounded, and presents many similarities with more recent work: probabilistic models, grid map, random evaders, persistent policy, *inter alia*. However, the algorithm is only evaluated with simplified simulations.

Markovian models: Hespanha's pioneer work has inspired much work in *probabilistic search*, a large part using Markovian models. Hollinger *et al.* consider a multi-robot team searching in a known environment for a moving non-adversarial target (Hollinger et al, 2009). They show that the resulting path planning problem is NP-hard, and that the optimal solution scales exponentially in the number of searchers. They provide approximation schemes and bounds on the performances. They use a POMDP formulation on the union space of the searchers' positions and the target position, but this may quickly become intractable for large environment or numerous searchers. Hence the robots coordinate implicitly: they only share their finite-horizon resulting plans (past and future paths), and do not plan for each other.

Hollinger's work also provides interesting results on complexity. The knowledge of the map indeed allows to provide bounds on the system performance. We think that one of the most important steps in this approach is the modelling step. Hence it is worth noticing that the maps and graphs are manually designed to limit the environment complexity, although the authors assert it could be automated. While they implemented part of the system on a Pioneer robot, the main results are only validated by simplified simulations.

POMDP is a powerful tool often used to model targets movements. In *probabilistic search*, as stated above, many work used Cartesian grid as a discretization of the environment (especially for the target possible positions), coupled with POMDP models for the evolution of the probability distribution. Ferrari considers the problem as a geometric optimization (Ferrari et al, 2009). She tries to maximize the probability of detection while minimizing the energy consumption, and consider fields of view constraints for the pursuers.

Yu *et al.* (Yu and Beard, 2011) couple dynamic models of winged AAVs with a grid decomposition of a Manhattan environment. They discretize and sample motion capabilities in order to handle the problem complexity, but they integrate some kinematics considerations in the motion models, although only simplified simulation results are provided, which minimizes the proven validity of such models.

Adversarial targets: Most of the Markovian models discussed above consider non-adversarial targets: it is globally easier, it better scales with large environments and numerous targets, and it requires less computing resources. Last but not least, it allows more global considerations: as targets "ignore" the pursuers, one can describe the probability distribution over time for a quite small cost compared to adversarial considerations. Although non-adversarial models are well suited to *Search and Rescue* problems, adversarial contexts exist (*e.g.* for some instances of *capture*).

Strom *et al.* notice this lack in probabilistic work (Strom et al, 2010), and propose to fill the gap between approaches considering adversarial targets with a sufficient number of pursuers as in (Vieira et al, 2009) in *capture*, and probabilistic approaches that consider non-adversarial targets (and lack of pursuers to perform a perfect sweeping of the environment) as in (Hollinger et al, 2009). Strom uses a parametrized semi-random target to model any kind of adversarial target. More precisely, he tackles the pursuit problem with a depth-first algorithm on a pruned tree. The original branching factor of the states tree scales exponentially with the number of pursuers and polynomially with the number of possible actions for both the pursuers and evaders. Inspired by Hollinger's work, Strom considers the problem as a rewards-maximizing search, and uses sequential planning and implicit cooperation in order to remove the exponential aspect of the branching factor.

The authors also consider semi-random targets to reduce the complexity: the targets sample their possible actions and choose the best ones. The number of samples is a parameter reflecting the target's skills or cunning (one sample = pure random ; when each possible action is sampled, the target is fully adversarial). However, what is the *best* sampled action is not obvious, and this aspect is not really addressed in the paper.

Surprisingly, the exhibited results show that the system handles quite well random walk based target as well as adversarial target or fixed planned target. Yet the system lacks several points, highlighted or not by the authors: one need to estimate the skill parameter of the targets, no communication issues are considered, and there is no consideration about the environment and the obstacles: they work directly on an abstract graph, as does Hollinger in (Hollinger et al, 2009).

They implement their system on wheeled robots and conduct experiments on a parking. The environment is simple (a small grid without obstacle), but a controversial assumption is about the space and time discretization: "an agent was only allowed moving one segment per time step". As discussed in Section V, this unrealistic hypothesis hides important issues about the integration of the system in real applications.

Particle filters: POMDPs are widely acknowledged as a powerful tool, but they hardly scale complex problems (Roy et al, 2005). Particle filters are a classical estimation technique to computationally handle a distribution of probability over a given space. They are most often used to cope with localization and tracking, but their application to a search algorithm is almost straightforward, and yields an advantage: one may use the same algorithm to find and then track the target, even if other techniques may be more suitable for tracking (see Section IV).

Mottaghi and Vaughan use a particle filter to perform a cooperative multi-robot target search (Mottaghi and Vaughan, 2007). They show that coordination improves the performance of the multi-robot system. The robots share their pose, and either their target observations or associated particles (which embed similar information). The coordination scheme is implicit: each robot takes into account the particles for which it is the closest. Thus the coordination scales linearly with the number of robots, as only data broadcasting is required. Each robot is then driven by a potential field method directly based on the particles it is in charge of, and a rough Dijkstra algorithm sets the direction and weight of the forces. The goal is to maximize the number of visible particles, thus minimizing their entropy.

The authors present simplified simulations and several tests in indoor environments with Pioneer robots; their experiments exhibit promising results. They fairly discuss the main drawbacks of their approach: among others, there is no optimality claim, although observed behaviours seem appropriate. Some dynamic loops may appear in the potential field because local minima. Finally, there is a strong full communication assumption (like in most of previous work). Besides, the system can easily handle several targets, and the motion model can easily be adapted and evolve dynamically. Still, like in previous algorithms, choosing a suitable target motion model is not trivial.

Particle sampling of the target position distribution of probability is also one of the way used by Riehl to dynamically update and reduce the search graph in order to deal with the associated NP-complexity (Riehl et al, 2007). This helps to address a cooperative *probabilistic search* problem, but also to extend the traditional dimension of the search space: for instance the work of (Riehl et al, 2007) handles both the path of the searchers and the position of gimballed sensors.

Decentralized systems: Most of the work presented so far are developed along a decentralized coordination scheme, and robots share either observations or individual plans. Most often little details are given on the implementation of information sharing, but it is a challenging problem to maintain coherence between decentralized observations of the evolving world. Bourgault, Wong, *et al.* study airborne search of stationary

or drifting targets (Wong et al, 2005). They consider Bayesian processes in a decentralized framework to estimate the current state of the world and keep this representation consistent between the team members. The system is assumed to be fully connected and the AAVs fly in obstacle-free areas; the authors give extensive results from simplified simulations. More recently, Cole *et al.* have used Decentralized Data Fusion (DDF) in a similar context (decentralized airborne observations), and they have put much effort in implementing the algorithm on-board real winged platform (Cole et al, 2010).

Decentralization schemes are often adopted for the sake of both robustness and scalability. Decentralized algorithms shrink the complexity from exponential to linear, most often through the renouncement for guaranteed global optimality. However, most papers assume the full connectivity of the network, which is not a reasonable assumption or imposes serious constraints on the systems, especially for ground platforms and in the presence of obstacles. Some papers do mention the issues raised by the communications constraints and the network connectivity, but hardly provide solutions.

Hollinger tackles the communication issue at the very first step of the conception of its algorithms, elaborating an implicit coordination scheme with a periodic connectivity (Hollinger and Singh, 2012). The coordination is sequential and linear with the team size: each team member plans on its own, according to other's current plans (see Figure 4). The authors highlight that the relaxation of the connectivity constraints allows varied and effective strategies, but it also decreases the robustness to failures. Their implicit coordination scheme is compared to market-based methods: the latter perform better, but at a much higher computational cost.

Hollinger and Singh show that multi-robot problems like the search problems are inapproximable when subject to connectivity constraints (Hollinger and Singh, 2010). In other words, "no polynomial-time algorithm can yield a multiplicative performance guarantee unless P = NP". This is a great argument in favour of decentralized sequential planning.
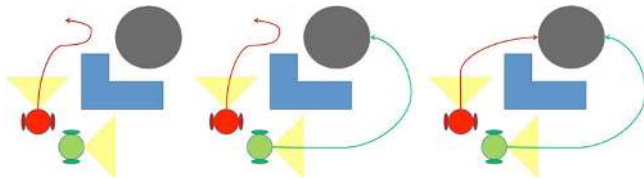
Fig. 4. Periodic connectivity search scheme *"The robots (green and red) must move around the obstacle (blue L-shape) to observe the area of high information gain (gray circle). They start in line-of-sight contact, and they must regain line-of-sight past the obstacle. The red robot first plans a path that remains connected to the green robots initial position (left). Then, the green robot plans a path that regains connectivity with the red robot past the obstacle (middle). Finally, the red robot replans to regain connectivity with the green robots new path (right)."* – excerpt from (Hollinger and Singh, 2012).

Optimization: While most researchers tackle *probabilistic search* as a planning problem, others formulate it as an optimization problem. The probability distribution indeed easily leads to a quantified space where optimization algorithms can be applied.

The optimization formulation is actually one of the oldest formulation of the problem, and was initially driven by the interest of the US Navy and US Coast Guard. One may refer to Stone's book (Stone, 2007) whose first edition of 1975 relates the early work in "optimal allocation of effort to detect a target", and partially refers to *probabilistic search*. The second edition published in 2007 highlights the results that have been found since then. As stated by Stone, "on the theoretical side, there has been significant progress made in solving the problem of optimal search for a moving target." (refer to the C-Appendix of the book). When considering stationary targets, new results are less significant.

Searching for moving targets, Ohsumi models the target as a Markovian stochastic particle, and El-Rayes uses a Brownian motion model to find the optimal strategy. Both consider a one *versus* one scenario, the former on a circle (Ohsumi, 1991) and the latter on a line (El-Rayes et al, 2003), These work gives theoretic guarantee but lacks tests in real conditions. One major limit of the models is that the functions are supposed to be continuous, which is hardly true for instance when considering visibility. Israel *et al.* provide a near-optimal algorithm that is able to handle such discontinuities (Israel et al, 2012).

Addressing the whole *probabilistic search* problem is complex, therefore some authors only seek near optimal solutions. For instance, Sarmiento *et al.* present a two-step approach, where they first consider a simplified search-path problem based on previously identified critical curves, and then refine this qualitative structure of the trajectory and locally optimize the path (Sarmiento et al, 2009).

Recently, Gan *et al.* use a decentralized gradient-based optimization to coordinate a team of AAV searching for target, with collision avoidance among the AAVs (Gan et al, 2012). As often with AAVs, and similarly to the other optimization-based approaches, the environment is considered as obstacle-free but for the roaming targets. Flushing *et al.*, on their side, formulate the problem as a well-studied MILP problem, for which efficient algorithms and implementations exist (Flushing et al, 2012).

Besides, as Stone highlights in his study, because of the computer developments "the trend in search theory is toward algorithms for computers and away from the theorem-proof style of presentation given in [his] book". This is illustrated for instance by the recent development of several swarm algorithms applied to multi-robot systems (Pugh and Martinoli, 2007; Hereford and Siebold, 2010; Ataei et al, 2013). If these algorithms show interesting results for multi-robot coordination in simulated environments, their application in the real world in not straightforward since they rely on strong communication assumptions (mainly because of the "pheromone-like" models). This issue is further discussed in Section V-A1.

Multi-Criteria Decision Making (MCDM): One of the main advantage of using optimization to solve robotic problems is that the solution can be built upon strong theoretic foundations. Amigoni for instance criticizes the too wide usage of *ad hoc* utility functions to evaluate candidates in robotic decision problems (Amigoni and Basilico, 2010). Indeed the proposed *ad hoc* utilities usually seem to provide satisfying results, but

they lack theoretical grounds and are not robust to changes in the expected behavior. Therefore Amigoni proposes to use Multi-Criteria Decision Making (MCDM) instead. Advantages are, among others, to deal more easily with Pareto frontiers when evaluating several criteria and to aggregate these criteria in a more proper and more meaningful manner, *e.g.* through Choquet integrals. It appears that MCDM effectively improves the performances through various experiments (Amigoni et al, 2013). This can "be explained by saying that MCDM better exploits the [information]" by traditional *ad hoc* utilities, as long as these information are precise and relevant.

Summary: Probabilistic approaches of the moving target search are quite recent, they rely on previous work on static surveillance and search for a static target. There have been plenty of promising work in the field, whose approaches are partially sustained by the current computation abilities. When worst-case guarantee cannot be provided because of the inherent complexity of *probabilistic search*, the algorithms are often built upon results from the applied mathematics community (on probabilities, game theory, MCDM, or optimization). Furthermore, the specificities of *probabilistic search* are closer to real world conditions than other classes of target detection problems (*e.g.* with respect to the lack of robots or the need to deal with priorities against time constraints in Search and Rescue scenarios). Probabilistic approaches may also ease the application of the algorithms on-board real robots as they represent well the uncertainties of the real world (Vidal et al, 2002).

D. Patrolling

While *probabilistic search* aims at going through an area looking for a target while giving some guarantees of finding it, *patrolling* is a similar mission, but with a perpetual, or at least cyclic aspect: the area is not covered once, but many times. According to the Oxford dictionary, to patrol is to "keep watch over (an area) by regularly walking or travelling around it": the notion of regularity is central. Though *patrolling* is related to *static surveillance*, *capture* and *probabilistic search*, optimal covering is not optimal patrolling (see Figure 5). We here mainly focus on area patrolling (2D), instead of position patrolling, which is a "pure" Travelling Salesman Problem (TSP), and perimeter patrolling (1D). As in *probabilistic search*, the goal is to "see" or detect a target: we do not address what the searchers must do after the detection in this section.

Fig. 5.   Optimal covering is not necessary optimal patrolling (inspired by (Glad et al, 2008))

Patrolling has been studied only during the last decade. Soon after pioneer work, Almeida *et al.* proposed a survey, defining

the basis for the future work on patrolling (Almeida et al, 2004). They propose an evaluation criteria to minimize, called *idleness*, which is the time elapsed between two consecutive visits of a same point or region in the area to patrol. This refers to the regularity highlighted above. Although other metrics have been proposed, like the performance against intruders of different kinds (Sak et al, 2008), the idleness and its variations (mean, worst, normalized, *inter alia*) still remain the main evaluation criterion in the related literature.

It has been observed experimentally (Almeida et al, 2004; Portugal and Rocha, 2011) and proved formally (Elmaliach et al, 2009; Pasqualetti et al, 2010) that the optimal solution against idleness is to solve a TSP over the whole area, in order to find an Hamiltonian cycle, and distribute the robots regularly along this path. Solving this problem is NP-hard in general, although the optimal solution may be found polynomially when facing specific topologies (Pasqualetti et al, 2010). This complexity, among other considerations, encourages to find other strategies.

Almeida also shows that randomness performs really badly, and that it is necessary to propose more elaborate solutions (Almeida et al, 2004). In a more recent survey Portugal *et al.* identifies several approaches (Portugal and Rocha, 2011). Below we follow, sum-up and briefly complete Portugal's overview.

Hamiltonian cycle: The TSP-based cyclic approaches aim at computing an Hamiltonian cycle in a topological representation of the area (graph), and to regularly place the robots along the cycle, all robots following the same path. Hence the solution is precomputed off-line, mainly because it involves solving a TSP, which is NP-hard. Elmaliach's work illustrates well this approach (Elmaliach et al, 2009). Although optimal when considering the idleness, this solution has several drawbacks. First, it lacks flexibility, it is not able to face dynamic environments or to handle constraints such as charging batteries. As stated by Acevedo, "The cyclic strategy theorically could reach the minimum maximal refresh time [...] but only for teams of homogeneous robots and assuming no communications constraints" (Acevedo et al, 2014). It is also centralized, and finding the Hamiltonian cycle can be challenging for some complex topologies or large graphs, even off-line. Polynomial time constant-approximation algorithms have been proposed to guarantee performances (Pasqualetti et al, 2010) but on average other approaches seem to outperform this guarantee (Almeida et al, 2004; Portugal and Rocha, 2013). The literature on TSP presents more efficient algorithms, but when the patrolling graph is not cyclic, the TSP is not a adequate solution on its own (Portugal and Rocha, 2013).

Alamdari *et al.* introduce priority levels on the various vertices of the graph. They try to minimize the maximum weighted latency, which is a weighted idleness criterion. and call this variation the *min-max latency walk problem* (Alamdari et al, 2013). The formulation first appears as more complex than the usual one, with a classical TSP underperforming on this problem as it does not handle the priority. They propose a new algorithm which "breaks" the complexity for each level of priority. Inside one level of priority, the classical TSP still performs better. Paradoxically, adding a new constraint (the

priority) helps to break the resulting complexity.

Besides its optimality, the Hamiltonian cycle solution is also independent of the number of robots. However, the reader may note that through all the work described here, no communication issue is considered, and the robots do not really cooperate in this context.

Moreover the Hamilton cycle solution may not be suitable for security scenarios involving the detection of a cognitive, adversarial target: the solution is indeed deterministic, and thus fully predictable: such patrolling schemes would be easily bypassed by aware intruders. This highlights the need for unpredictability, through randomness and sub-optimality, which drives other approaches, and justifies the usage of alternative, complementary metrics (Sak et al, 2008).
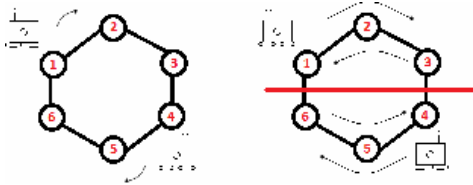

Fig. 6. Example of an Hamiltonian cycle strategy (left) and of a partition-based strategy (right) – excerpt from (Portugal and Rocha, 2011).

Partitioning: The most common alternative to the optimal Hamiltonian cycle, which solves the problem as a whole, consists in partitioning the area into sub-regions, and assigning one agent to each region (see Figure 6). Variations exist in the way to split the initial area, and how to define the patrol scheme of the subregions. Generally, the area is modelled as a graph, and the approach relies on the large literature on graph partitioning. One may also use path partionning strategies, but this is less efficient than the area partionning schemes (Acevedo et al, 2014).

Among others, Portugal *et al.* proposes the *Multilevel Subgraph Patrolling (MSP)* algorithm to split the graph (built upon a grid-based representation) and define the local optimal patrolling path for each robot (Portugal and Rocha, 2010). Experimental results show that this algorithm performs well, with performances close to the optimal. However, the drawback of determinism still remains.

In (Fazli et al, 2010), Fazli's approach relies on the traditional *static surveillance* or *art gallery* problem to define the waypoints underlying the topological representation of the area (graph). A Constrained Delaunay Triangulation (CDT) is used to build a suitable topological map, instead of building the graph directly from a grid. This aims at taking into account the limited sensors range of the agents. Then the patrolling cycle of the robots are built upon spanning trees. Guarantees over the complexity and the completeness of the algorithm are provided[5].

When facing non adversarial targets, the problem is also referred to as a *persistent monitoring* task. Soltero proposes to solve this problem with a partitioning approach based on the

optimisation of a given cost function (Soltero et al, 2013). This results in a command law for the robots that takes into account various time-invariant levels of interest in the area. It is worth noticing that there is no graph structure, and the results are valuable for continuous models (although for computational reasons one may discretize the functions). Soltero's approach has been extend by Kuhlman *et al.* with the addition of a waypoint feedback policy, taking into accounts obstacles, collisions and motion uncertainties (Kuhlman et al, 2014); this extended approach is used in a maritime context.

Multi-Agent Systems: Another kind of approaches relies on classical Multi-Agent Systems (MAS) mechanisms. Auctions algorithms are one solution to solve the *patrolling* problem quite efficiently, although the usual limits stand, especially those related to communication issues. They usually provide interesting features as scalability, robustness, decentralization and flexibility. One may refer to Portugal's survey for more references (Portugal and Rocha, 2011).

The same authors recently proposed a distributed MAS approach to the problem (Portugal and Rocha, 2013). They argue that for scalability purposes, one need to minimize interferences between agents. They use Bayesian-based techniques, and compare their algorithm to prior work: it seems indeed both scalable and robust to failures, and shows interesting performances, including an implementation on-board real robots in an indoor environment. One may regret the lack of statistical analysis on the performances, but two interesting results are worth to notice: first, the implementation of the Hamiltonian Cycle approach does not show the best performances, and this could be expected when one consider the non-cyclic topology of the graph used for testing[6]. Second, the implementation of the partitioning algorithm performs best, especially when the number of robots grows (above ten). This highlights that topological considerations and team composition should significantly drive the choice between all existing algorithms.

In a different context, Raboin *et al.* used market-based to guard an maritime asset and its surrounding area (Raboin et al, 2013). The specificities of their approach lies in the three different tasks for the ASVs[7]: observing, guarding, and intercepting. Indeed, their patrol schemes are oriented towards the potential intruders and the uncertainties in assessing if targets are actual foes or only neutral boats passing-by. It is also worth noticing that their objectives differ from the traditional idleness: they interestingly assume that the intrusion cannot be preventde but only delayed, and try to maximize the expected minimum arrival time of the intruders next to the guarded assets.

Marker-based strategies: Several work use markers in the environment to communicate data between agents and coordinate the latter (Almeida et al, 2004; Chu et al, 2007). Among them, swarm approaches are inspired from the ant pheromone process and consist in having a grid-based virtual representation of the environment, in which the agents can depose and sense pheromones. This allows to guide them

---

[5]Note that the article does not refer to *patrolling* explicitly, although it exactly embraces its scope.

[6]The graph being non-cyclic, there does not strictly exist an Hamiltonian cycle: one should "re-use" some vertices to cover the whole graph, hence the non-optimality.

[7]Autonomous Surface Vehicles.

locally, without great cognitive capacities as the intelligence rely in design, the interaction between the agents and the environment, and the information propagation. It prevents any combinatory explosion, and handles dynamic changes. It is worth noting that resulting solutions appear to always converge to an Hamiltonian cycle or a combination of Hamiltonian cycles (Glad et al, 2008).

In (Glad et al, 2010), Glad *et al.* develop some of the main problems raised by swarm approaches and their implementation on real robots. This gathers synchronous / asynchronous considerations, collision between agents, the requirement for local communication, space and time granularity, interference, non-determinism, *inter alia*. They show that marker-based strategies are viable, but raise numerous underlying issues during the integration process.

Learning: Another way to solve the *patrolling* problem stands in learning techniques. They are mainly used in relation with Markovian formulation, and the classical Q-learning. The idea is to learn local strategies that the agents may re-use, while adapting to other agents, to the target, or the dynamism of the environment. Ruan *et al.* have adapted the classical learning to introduce less predictive behavior (Ruan et al, 2005). They partition the area, and for each subregion they compute off line a set of routes, using MDP and Q-learning. Then agents choose on-line between these routes at each cycle. To avoid the deterministic aspect of learning, the action selection process (softmax) allows agents to choose their action between a set of actions according to some given distribution of probability.

Game strategies: In an adversarial context, one may naturally tend to consider Game Theory. This is particularly suitable when one wants to consider the targets' behaviour. Amigoni *et al.* propose a game-theoretic approach (Amigoni et al, 2009), in contrast to randomized unpredictable strategies. Their argument is that modelling the adversaries allows elaborating better strategies, although these latter would be sensitive to the quality of the model and the assumptions. In a mono-robot context, Amigoni models the agents as a 1st-order Markovian process. Discretization is crucial both in space and time, and the intruder is assumed to take some time to penetrate or perform its goal. As the intruder chooses when to attack, Amigoni proposes a leader-follower equilibrium, *i.e.* a strategy for the patroller that minimizes the impact of the intruder reacting to the patroller moves. Game theory brings numerous properties for this equilibrium, which allows to define the suitable strategy. The approach is mathematically grounded, but lacks realism in the hypotheses, and scales badly with the dimension of the environment.

Although slightly out of the scope of robot or autonomous systems, it is worth mentioning Pita's work which has been in used for months as a decision-making tool to define patrolling schemes at the L.A. Airport (Pita et al, 2009). Pita's paper highlights the two mains issues in adversarial patrolling: the lack of resources which enforces the patrol schemes (full coverage is not possible), and the need for randomness, as vulnerability comes from predictability. He uses a Bayesian Stackelberg game to model and elaborate the patrolling schedules of the agents. The bayesian formalism helps to handle uncertainty about the adversary, while Stackelberg game theory gives optimal random strategies. The framework takes into account the priorities in the randomization (important areas) and is able to deal with mixed-initiative[8]. This is operationally useful, as sometimes the systems may be unaware of real-world constraints. The usage of the Armor systems helped the L.A. Airport Security to provide some quality guarantee on the schedule, as it is known that human barely deal with true randomization (which is crucial for security). Therefore the system helps to "randomly" setup checkpoints and choose where to allocate specific resources (like canines). It is not made for robotics systems, hence no path is given, and the number of both actions and places are limited, but the system provides a way to solve NP-complete problems of this kind, and was successfully tested in an operational context for months.

Perimeter patrols: Patrolling over an area involves at least 2D considerations, although higher space may be desired (in a 3D context or with steerable sensors for instance). However, the perimeter patrol is also a classical one-dimension patrolling problem. Agmon *et al.* address the multi-robot perimeter patrol in adversarial settings patrol *around* an area (Agmon et al, 2008), preventing the penetration of an area. They provide a non-deterministic solution, highlighting that "in adversarial settings the frequency criteria becomes less relevant." The 1D model subdivides the perimeter into segments, and tries to minimize the probability of penetration for every segment against a cognitive and observing intruder. The authors study several strategies, and give mathematical statement and guarantees, but do not provide experimental results. The reduced dimension allows considering a constrained problem and find an optimal solution more easily than in the case of area patrolling.

Summary: Patrolling is a rather recent field of study but numerous approaches have already been proposed, along with various benchmarks. It is also strongly linked with other problems described in this survey and can rely on a large theoretic framework including game theory and graph theory. Graphs are indeed the predominant model of the world, and this aspect of patrolling, embodying the world topology into a graph, is common to previously mentioned classes of problems as *static surveillance* or *capture*. This may explain how fast many interesting results have been proposed for patrolling. Although there is a lack of realistic implementation to validate the proposed algorithms, many theoretical results have been shown. Moreover, and as often when facing an NP-hard problem, suboptimal solutions are welcomed as they present interesting features such as robustness, flexibility, and security in adversarial settings. Recent work tend to emphasize these considerations, and to study local algorithms while providing guarantee on the optimality.

E. Hunting

The class of the *hunting* problems tackles the target detection without providing any guarantee. This lack of guarantees generally comes from a lack of means, be it the number of

---

[8]Mixed-initiative interactions allows users to occasionally adjust or override the automated schedule.

robots, the quality of the sensors, or the information about the environment or the target.

No map available: The absence of map may come from a lack of initial information or from a design choice. Working without any map — except local perception of course — indeed allows rather robust strategies, yet rarely efficient. It is a trade-off between performance and robustness to error and failures, with the assumptions and the amount of information as a cursor. The design may also be driven by the poor quality of the sensors. This search problem without global map is related to exploration, especially when targets are stationary – but exploration strategies are out of the scope of this survey.

Cao uses distributed control and only local interactions ("Local Interaction with Local Coordinate Systems", LILCS) to perform a *hunting* task in an unknown environment (Cao et al, 2006). His robots perform a random search locally coordinated to pursue and catch the target. The system is quite reactive and robust to accumulated drifts and communication failures.

Annas and Xiao pursue a target in a unknown 2D environment without any motion model for the target (Annas and Xiao, 2009). They try to locate and then capture the target (the second part is similar to *following*, see Section IV-B). The evader also has a limited knowledge about the environment, and they have validated their approach against human players (Annas and Xiao, 2010).

Kolling and Carpin perform graph-clearing-like missions without any prior map, and no map is built during the process (Kolling and Carpin, 2010a). They seek optimality but without global coordination: the approach is decentralized, locally frontier-based, and the pursuers sweep the environment gathered as a front line (because of their limited field of view), following walls or nearby robots. The explored areas are guaranteed to be cleared from any target, but there is no guarantee that the whole environment will be cleared, even with sufficient resources, because of the local decision process and the absence of a map memory.

Between uncoordinated random strategy and fully coordinated sweeping strategy, Miao proposes an emergent motion strategy, based on self organization (Miao et al, 2010). Most of self organization models are flocking ones, biologically-inspired, but they are not suitable for *hunting* as the agents stay close together. On the opposite, Miao proposes an *anti*-flocking algorithm that mimics solitary animals social behavior, like tigers or spiders. His system shows similar performances in simulation with a fully coordinated sweeping strategy while being more robust (no communication is required for instance).

Lack of resources: Raboin addresses the search problem without enough robots to perform area-clearing and any without probabilistic consideration (Raboin et al, 2010, 2012). Relying on an information theoretic background, he proposes geometry-based and game-theoretic algorithms, but with imperfect information (whereas such algorithms usually rely on perfect information), and using an heuristic based on a relaxed version of the problem. His latter work extends the results from a grid world to a continuous world (namely a partially observable Euclidian space) and takes into account the communication failures. His work is also related to *following*

as the target "frequently passes in and out of visibility", but the initial target location is not known, and what really matters is to find this target location.

When one considers *probabilistic search* in a cluttered 3D environment, the overall problem seems intractable because of its complexity (the 2D problem being NP-hard at best). Dornhege *et al.* show that despite this intractability, one may come up with a computable solution of reasonable quality (Dornhege et al, 2013). They try to cover a predefined 3D area with one robot, in a way similar to an *Urban Search And Rescue* scenario where time matters and where one cannot wait for the optimal solution to be computed. Considering that the problem is too hard, they try to provide a near-optimal solution at a small cost. The algorithm has two steps: first they try to find a minimal set of 3D poses and sensor orientations (called *views*) that cover the area, and then apply a TSP on this reduced problem to define the order of visit between the views. The authors compare their results to a full planning approach (which ends up with the optimal solution, but at a high cost): the performances of the algorithm are very close to the optimal while being ten time faster to compute, and perform better than a greedy approach. This assertion is actually only partially true, as only the execution time is considered, and not the sum of the computation time plus the execution time, for which performances seem to be similar.

Summary: The specific class of *hunting* problems is not as much addressed as the graph-clearing or the *probabilistic search* instances of the mobile search problem. Most of the work presented in this section do not present any thorough search strategy, as they mainly rely on randomness, but they provide interesting insights for who lacks information, robots and/or sensors quality. The most recent work on the field still keep this restrictions and introduce new interesting strategies that cope with such limitations. In both cases, the main interest is robustness and cheap platform integration, in terms of required work and of platform cost.

IV. TARGET TRACKING

In the following sections, we review the contributions to the different classes of problems defined in the *target tracking* (right) branch of the taxonomy. Again we gather work from various communities (control, decision, applied mathematics, multi-agent systems,), to highlight the main strategies and how complementary they are.

A. Target Localization

target localization involves a team of mobile sensors tracking targets through multiple and simultaneous observation on each target. Multiple points of view add extra information on the target and mainly aims at improving the precision of its estimated position. *target localization* is also designed in the literature as the *Focus of Attention* (FoA) problem (Isler et al, 2005) or the *Cooperative Localization and Target Tracking* (CLATT) problem (Mirzaei et al, 2007). First contributions appeared in the early 2000's (Isler et al, 2005; Cowley et al, 2004; Grocholsky et al, 2005), and the problem has gained a growing attention since (Xu et al, 2012; Zhou and Roumeliotis,

2011; Moseley et al, 2009; Tsokas and Kyriakopoulos, 2011; Santos and Lima, 2010).

Cooperative context: Cooperation is at the heart of *target localization*, for here one wants to take advantages of multiple points of view and sensors. One may benefit from a heterogeneous team composition: besides sensors differences, the cooperation of AGVs and AAVs in the same team (Grocholsky et al, 2005; Moseley et al, 2009; Xu et al, 2012) enables complementary accessibilities in the environment and a variety of vantage points (Figure 7).
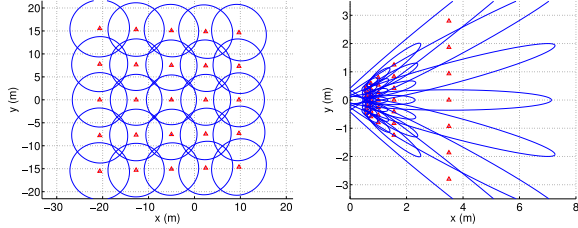

Fig. 7. " *Ground feature observation uncertainty from aerial (left) and ground (right) vantage points. The AAV camera looks down* 5° *off vertical at 50m altitude. The AGV camera is mounted horizontally 0.32m above the ground plane. Comparative feature observation accuracy is illustrated by ground plane confidence ellipses associated with uniformly spaced pixels in the imagery.*" – excerpt from (Grocholsky et al, 2005). This highlights the potential of a heterogeneous mobile sensors team.

Several work rely on centralized approaches to determine the next best location for each sensor (Mirzaei et al, 2007; Zhou and Roumeliotis, 2011). Isler defines a variant of *target localization* called *Focus of Attention* (FoA), and shows that this is a NP-hard problem difficult to approximate (Isler et al, 2005). Therefore he studies several specific constrained geometric cases (lines and circles) to give good approximation results. Strongly grounded by mathematic results, his centralized approach provides proofs and bounded guarantees of the proposed approximate solutions.

Besides these centralized approaches, Moseley uses a mixed centralized-decentralized method (Moseley et al, 2009), stating that full decentralized systems are mandatory when a large number of robots are involved ("tens to hundreds"), whereas centralized approaches allow optimal performances and seems affordable for a small teams of robots. Several authors propose distributed approaches (Cowley et al, 2004; Grocholsky et al, 2005; Kamath et al, 2007; Xu et al, 2013), and often deal locally with the cooperation aspect.

Nevertheless *target localization* is not just about the motion of the robots respectively to the target, but also about data fusion, as one expect to take advantage of the multiple information about the target. Data coming from various sources need to be merged to properly exploit the multi-robot synergy. Numerous methods have been proposed to perform data fusion in this context: extended-Kalman Filters (EKF) (Mirzaei et al, 2007), Decentralized Data Fusion (DDF, similar to a decentralized linearized Kalman filter) (Nettleton, 2003; Grocholsky et al, 2005; Moseley et al, 2009), Rao-Blackwellised particles filters (RBPF) (Qian et al, 2008), Multiple Hypothesis Tracking (MHT) (Tsokas and Kyriakopoulos, 2011) or filters on Gaussian Mixtures Models (GMM) (Santos and Lima, 2010).

Integration issues: As stated by Cowley (Cowley et al, 2004), the communication issues and the distributed database context are crucial when one addresses the problem on teams of real robots. This certainly explains why most of the work presented in this section take after work in distributed computing and distributed database management. Cowley highlights the huge gap between the data size (multiplied by the number of robots) and the communication flow which stay low (and may even be reduced by the number of involved robots). This raises the interest of distributed database and of sharing summarized, processed data. One may also try to reduce the need for communication: using learned utilities, Xu *et al.* allow implicit coordination to drastically reduce the data flow for negotiations (Xu et al, 2013).

It is worth noticing that for *target localization*, there have been many efforts to integrate the algorithms on real robots (Grocholsky et al, 2005; Moseley et al, 2009; Santos and Lima, 2010; Mirzaei et al, 2007; Xu et al, 2013; Tsokas and Kyriakopoulos, 2011), although the reader must also pay attention to the scope of the experiments (both in simulation and with real robots) that do not always reflect what is presented in the algorithms or are not validated in realistic settings (Kamath et al, 2007).

Variations: Target Localization may also concern several targets at the same time (Kamath et al, 2007; Isler et al, 2005; Tsokas and Kyriakopoulos, 2011). In such cases, the problem is related to *observation* (see section IV-C), where the difficulty is also to allocate targets between the observers.

Another variation is to consider a changing target, or more specifically, a target whose shape changes: this may be a forest fire, a chemical spill, a flock or a mob. The target has here some specificities that may be taken into account to specifically design more efficient algorithms. For instance, Casbeer and Kingston tackle forest fire monitoring with AAVs in a distributed manner (Casbeer and Kingston, 2006), while Clark and Fierro perform perimeter detection and tracking for various substances like chemical spill (Clark and Fierro, 2007). Note that due to the specific nature of the target, this work is also related to the barrier *static surveillance*, where the robots "guards" the boundaries of the target (see Section III-A). Clark's algorithms are also related to *hunting* (see Section III-E), since at the beginning the robots randomly search for the substances in the environment. This is an additional illustration of the strong links between the left and the right parts of the taxonomy.

Optimization: Target Localization can be formulated as an optimization problem. For instance, Kamath *et al.* formulate their problem as a well-studied energy minimization problem and propose an iterative and distributed algorithm to solve it (Kamath et al, 2007). Xu considers the cooperation problem as a traditional Mixed Integer Non-Linear Programming (MINLP) optimization problem (Xu et al, 2012). For this kind of problem only centralized solvers exist, this is why Xu makes approximations to reduce the problem to a non-linear constrained optimisation problem, which can be solved in real-time by a decentralized system.

Summary: The *target localization* problem is difficult by several aspects, be it sensor positioning relatively to the target

or the multi-robot cooperation issues (task allocation, data fusion, communication, *inter alia*). For this last part, many researchers rely on prior related work in specific communities (among them: sensor network, information theory, control, distributed systems) and focus on the *target localization* applications. Centralized approaches may give better results and better performance boundaries than decentralized ones. But they are not as realistic as the decentralized approaches and the problem complexity prevents on-line optimal solving. Moreover distributed approaches are often integrated and tested on-board real robots, which allows a greater approval of the conception choices.

B. Following

Early work: By *following*, we refer to the problems of tracking a single target with a single robot. These kind of problems have been largely studied, with a strong emphasis on the mathematical point of view (geometry) (Chung et al, 2011; Alexander et al, 2009; Nahin, 2012). Early work refer to it as the traditional pursuit problem, or as the *Lion and Man* problem, and were also motivated by naval conflict scenarios – the reader may refer to the introduction of (Noori and Isler, 2014) for a good state of the art on the *Lion and Man* problem.

The general consideration is that going straightforward to the target is a naive strategy; the pursuer must take into account the obstacles to plan an optimal move (Murrieta-Cid et al, 2002; Bandyopadhyay and Ang, 2006). More generally, visual servoing approaches to target tracking (as in (Papanikolopoulos et al, 1993)) have limited success because they do not take into account the complexity of the environment. Geometry-based algorithms go beyond these limitations, and this section focuses on these approaches.

Contrary to visual servoing, a too abstract formulation may lead to results not applicable on real scenarios. For instance, in 1962 Eaton and Zadeh (Eaton and Zadeh, 1962) use Markovian models to elaborate the best strategy to catch a fixed target and then generalize the result to a moving target. But the approach requires many parameters and mathematical functions which are not easy to relate to the real world, and so the results are somehow ethereal.

Visibility constraints: The tracker must keep in sight the moving target, and so visibility constraints must be considered. First because continuous target visibility is often a requirement, and second because losing visibility implies a risk to never recover the target. Both the obstacles and the sensor range constrain the pursuer visibility and strategy. LaValle considers a discrete environment and tries to plan an optimal path to maximize the visibility over a predictable target (LaValle et al, 1997), while Murrieta-Cid proposes a reactive planner to maximize visibility over an unpredictable target (Murrieta-Cid et al, 2002). The latter also introduces the notion of *shortest distance to escape (SDE)*, which is similar to the notion of vantage time and gap-zone defined by Bandyopadhyay and Ang (Bandyopadhyay and Ang, 2006). The gap-zone is determined by the surrounding obstacles and leads to a greedy pursuit strategy. The authors show how this strategy outperforms a naive greedy "run to the target" strategy. Implemented on-board a robot, the pursuer is able to follow a human being through a crowded building (cluttered dynamic environment) in real-time.

More recently, visibility constraints have been added to the classical *Lion and Man* problem in (Noori and Isler, 2014), where the problem is formulated with finite field of view for the pursuers. The authors notably show that "a single deterministic pursuer with line-of-sight visibility can capture an evader whose speed is equal to the pursuer's in any monotone polygon". Note that capture is no longer guaranteed when one slightly relaxes the monotonicity constraint by considering the class of weakly monotone polygons.

Complexity result and locally optimal strategy: More recent work by Bhattacharya study the atomic local configuration composed by one pursuer, one evader, and one polygonal obstacle, considering a continuous environment (Bhattacharya et al, 2007). The authors distinguish different areas in the geometric decomposition of the scene, each area corresponding to a specific optimal strategy (see Figure 8). Karnad and Isler proposed similar results for a circular obstacle, and given any initial position can determine which protagonist will win the game (Karnad and Isler, 2009). This approach has a drawback: it barely scales with the number of obstacles. Defining an optimal global strategy for the whole *following* problems has been proven to be NP-Complete: in (Murrieta-Cid et al, 2008) the authors show that there is an underlying *Traveling Salesman Problem (TSP)* in *following*, and the *TSP* has already been proved to be NP-complete.

(a) Pursuer-based partition   (b) The geometry of the partition

Fig. 8. Bhattacharya geometric pursuer-based partition of the environment. Considering the obstacle, the evader position and the difference of speed between the evader and the pursuer, the authors partitioned the environment into different areas where optimal strategies for the pursuer are defined and outcomes of the given strategies are predicted – excerpt from (Bhattacharya et al, 2007).

Facing this complexity, Karaman and Frazzoli propose an incremental sampling based algorithm for the evader strategy, using on numerical methods instead of analytical ones (Karaman and Frazzoli, 2011). Indeed, "although analytical solutions of some simple pursuit-evasion games are known, most interesting instances can only be solved using numerical methods requiring significant off-line computation." The algorithm has anytime properties, and the authors provide demonstrations of its probabilistic completeness and soundness guarantee.

Dynamic constraints and environment complexity: Recently, several authors have specifically considered the dynamics of

the pursuing robots, which constrains their motion abilities. Difficulties appear indeed when there are differences between the target motion restrictions and the robot motion restrictions. Dynamic considerations become more frequent for winged AAV (Theodorakopoulos, 2009) (although the models are still somehow simplistic), but they remain rare for AGV. However, these considerations matters and have a great influence on the resulting strategies. The reader may refer to Murrieta-Cid's work for instance, which proposes a pursuit strategy for a differential driven robot following an omnidirectional target (Murrieta-Cid et al, 2011).

Besides the complexity introduced by the models of the protagonists, models of the environment also have an underlying complexity. As a matter of fact, most of the authors consider a 2D environment, where obstacles stand both for the vision and the motion, and are shared by both the target and the pursuer. But in actual 3D environments, more difficulties arise, like buildings or undergrowth for AAVs: visibility contraints are not equivalent to traversability constraints. More generally in real applications, obstacles for motion are distinct from obstacles for vision. Recent efforts have been made to consider more complex environments: Bhattacharya tries to reduce the 3D problem to 2 dimensions by projection (Bhattacharya et al, 2007), and more recently perform actual 3D simulation (Bandyopadhyay et al, 2011), while Robin and Lacroix consider 2.5D multi-layered models for visibility and motion (Robin and Lacroix, 2012). Obviously, embodying a richer environment expressiveness has a cost: the extension of the problem dimension. That may be a severe issue as the considered problems are already complex by their combinatorial nature.

Besides the need to rely on realistic models, more abstract models than the classical 2D Euclidian world have an interest: the historical *Lion & Man* problem still motivates mathematicians, and recently Alexander (Alexander et al, 2006, 2010) extends some results in Euclidian pursuit-evasion to any CAT(0) space[9] (and even CAT(K), $K > 0$ spaces). If non-Euclidian CAT(0) space i familiar to many roboticists, the authors mainly argue that most 2D (Euclidian) results can be generalized to CAT(0) spaces, and that sometimes finding a solution in the latter may be easier (*e.g.* polynomial instead of NP-hard) as the space is less constrained. However, no actual integration on-board robots nor simulation results were provided.

Target models: As stated before, most work presented here assume the target has the similar motion abilities as the pursuer. The target's behaviour is often considered as non-deterministic, which does not prevent planning under a reasonable temporal horizon as the behaviour remains independent from the pursuer's state (Robin and Lacroix, 2012). It is also frequently assumed that the target may be guided by obstacles and its current heading (Bandyopadhyay and Ang, 2006). PMODP formulation may also embody the target's behavior (Bandyopadhyay et al, 2009), although the difficult

computability of POMDP constrains the size of the environment and the considered time horizon.

Variations: In the traditional formulation of the *following* problem, the pursuer tries to "catch" the evader, *i.e.* to get as close as possible to the target. A variation of the problem consists in maintaining a bounded distance between the pursuer and the target (Murrieta et al, 2004). Indeed there exists many cases where one wants to maintain this distance, *e.g.* for safety reasons or to have a better observation/visibility on the target.

Inspired by the piano movers problem, Murrieta-Cid *et al.* define sufficient conditions to escape a finite sensor range pursuer, thus determining an evader strategy (Murrieta-Cid et al, 2007). Similarly Bhattacharya provides necessary and sufficient conditions to break the visibility link between the robot and the target, and defines game strategies for both players leading to a Nash equilibrium (Bhattacharya and Hutchinson, 2009).

The problem of maintaining a certain distance with a good visibility (point of view) on the target has a wide application field in computer games where one desires to maintain visibility on the characters in 3D virtual environment – (Oskam et al, 2009) is a good introduction on the matter. The quality of the camera navigation is crucial for a good experience of the game. Even if the constraints are different from robots world (intercuts are allowed for instance, which would be similar to teleport robots), there exists similar challenges, which may inspire solutions for robotics.

It is worth mentioning that less usual problems are also tackled. For instance, Bandyopadhyay *et al.* study stealth tracking: the goal is to visually track a target among obstacles while remaining hidden from the target sensors (Bandyopadhyay et al, 2005). This is related to a recent field of robotics problems: *covert robotics* (see Al Marzouqi for a recent survey on covert path planning (Al Marzouqi and Jarvis, 2011)).

Vo and Lien take advantage of the special features of coherent targets (mob, swarms, *etc.*) to find and track them among the environment (Vo and Lien, 2010). The main feature of coherent targets is that they require a minimal space to hide, but have no predefined shape. With a slightly different purpose, Wang *et al.* track a target with a whole flock of robots (up to sixty robots in simulation) (Wang et al, 2010). It is still a following problem, as the flock is considered as a single entity, although its individual components have their own local behaviour, and the tracking is achieved through a consensus process between the agents. This may be used to track a target or to follow some trajectory for the whole flock.

Finally, and as for the other classes of problems, *following* may occur in maritime contexts. Besides the impact on the motion, similar strategies are at stake – especially global and optimal strategies are not tractable in real time. One may find in (Svec et al, 2014) a good introduction of *following* problems in maritime contexts.

Summary: In brief, *following* and its variations have been widely studied for decades and many approaches have been provided. The literature also refer to them as *pursuit-evasion* problems or as *Lion & Man* problems. The intrinsic nature of the *following* problem promotes geometric approaches, be they local or global. Finding a global optimal strategy is NP-

---

[9]In a CAT(k) space, k is a lower bound of the space curvature. A notable special case is k = 0: complete CAT(0) spaces are known as Hadamard spaces. Traditional Euclidian spaces are CAT(0) spaces.

complete and thus not tackled in real conditions nowadays, but many reactive strategies have been proven quite effective, both in simulations and on-board robots.

C. Observation – CMOMMT

Early work: Observation (or CMOMMT, see Section II) aims at simultaneously observing several mobile targets with a cooperative team of robots. Considering the difficulty of the approach (the problem is NP-hard), Parker and Emmons proposed a distributed and approximated approach named A-CMOMMT (Parker and Emmons, 1997). They use the ALLIANCE formalism to achieve high level control, and force fields for low-level control. The high-level behavior-based architecture activates or disables force fields to achieve proper observation motions and avoid local minima. The algorithms have been tested on-board actual robots (Parker, 2002).

Jung and Sukhatme propose a region-based coarse approach (Jung and Sukhatme, 2002): inside each region, the observers use a local strategy to maximize the number of observed targets. No actual negotiation occurs among the robots. Again, the algorithms have been tested on both simulation and on-board robots. Two points are worth mentioning: first, real robots experiments gave worse results than simulation experiments, because conditions were worse (narrower and shorter field of view, worse odometry, *inter alia*); second, the authors have tested different team compositions with both fixed and mobile sensors (even if they do not tackle the fixed sensors placement issues). They highlight that fixed sensors are beneficial, especially when coordination between the robots are bad. In this latter case, a mixed team or even only fixed-sensors composed team performs better than mobile sensors only.

Communication and active cooperation: Kolling proposes a more recent distributed behaviour-based approach, named B-CMOMMT, where robots are responsible for the targets they monitor, but may ask for help from others when needed (Kolling and Carpin, 2007). This allows more active cooperation and appealing cooperative behaviour like targets exchange between robots. More important, the framework is able to overcome situations where the older A-CMOMMT would fail, through active communication and cooperation, without any loss of performance for other situations. Only (but numerous) simulation results are given.

Derenick *et al.* present a discrete-time optimization framework for the problem of tracking (observing) several targets with multiple cooperating robots (Derenick et al, 2009). They use interesting mathematical results from the spectral graph theory to provide performance guarantees on both full target coverage and network connectivity. Unfortunately they only tested the algorithms on simplified simulations with no obstacles: this limits the current validity of this approach.

Tang and Ozguner also formulate *observation* as an optimization problem: with a team of winged AAVs, they try to minimize the average time elapsed between two consecutive observations of each target (Tang and Ozguner, 2005). Focus is on the kinematics of the agents and the motion control strategy. They first consider a mono-robot instance of the problem, before extending the algorithms to a cooperative strategy.

Variations and relation with other problems: In a more environment-centered approach, Pimenta *et al.* address the problem of *Simultaneously Covering an area And Tracking intruders (SCAT)* (Pimenta et al, 2010). They actually cover the area with a Voronoi tessellation of the map, and detected targets influence the density functions on which the tessellation is based – which highlights the links between the detection and tracking parts of the taxonomy.

The observation phase may indeed be preceded by a detection phase (Parker, 2002; Kolling and Carpin, 2007), where robots try to find the targets, *i.e.* perform a *mobile search* or a *static surveillance* of the area as defined in the taxonomy.

Observation is also related to the other classes of tracking problems. For instance, when the precision of the targets location matters, *observation* may be linked with *target localization* (Kamath et al, 2007; Isler et al, 2005) (see Section IV-A for more considerations on the related work).

In a similar way, as presented in Section IV-B, Vo considers a monitoring problem with multiple targets, but his approach is closer to *following* as the focused targets are put together into a flock, and considered are one single target whose shape may change (Vo and Lien, 2010).

Markov and Carpin tackle *observation* (CMOMMT) from the point of view of the evaders (Markov and Carpin, 2007). More specifically, the evaders actively cooperate through communications to share data. This awareness allows to minimize the average observation time through a force field approach which embeds several basic principles for the evaders, like "stay unseen as long as possible", "stay scattered but preserve the communication links", *inter alia*.

Summary: The *observation*/ CMOMMT problem is far more recent than the *following* problem, and its development goes with the growing enthusiasm for multi-robot problems and approaches. It shows that global performances of multi-robot teams are better than several one vs. one instances performed in parallel, *when* there exists any active cooperation between the team members. Otherwise, no performance improvement can be ensured; worse results may even occur.

V. SYNTHESIS

Through the variety of problems reviewed along this survey, there are many comparable aspects: similar or recurrent approaches, models, assumptions, validation processes, flaws, *inter alia*. Here we try to highlight the ones which we consider important, namely the world and agents models, the main approaches, the current trends, and the validation processes. This part relates to both sections III and IV, as similar platforms, formalisms and models are common to both the detection and tracking problems. In actual scenarios, the tracking may indeed directly follow the detection phase, or be performed in parallel: in a multi-robot context, once a target is detected, some robots track it while the others continue the detection task. There is a large variety of models and approaches, but in the end it is the results and the validation processes that assess their validity *a posteriori*: we also discuss these validation processes for the reviewed work. Finally, we define and present what we consider as the three main open research directions on

which the research communities should maintain or pursue their efforts.

A. Models

Models that represent the environment and the agents (the robots and the targets) are at the core of the decision processes, and are required whichever the chosen approach. The combination of environment and agents models yields the ability to *predict* the outcome of the agents actions, *i.e.* what the agents are able to do and the expected consequences of the possible actions.

The reader may have already noticed that most of the encountered models are rather simple, and in our opinion often too simple. They present a limited expressiveness, and if they ease the way to solve the problem (somehow by reducing its complexity), they are a too coarse abstraction of the reality, hardening the implementation issues, and thus minimizing the realistic validation of the proposed approaches.

1) The environment:

a) Beyond 2D models:: Most authors use a 2D single-layer representation of the world, be it a grid (Yu and Beard, 2011; Flushing et al, 2012; Robin and Lacroix, 2012) or more continuous models: triangulation (Fazli et al, 2010), Voronoi tessellation (Pimenta et al, 2010), specific tessellation (Bhattacharya et al, 2007; Murrieta-Cid et al, 2007) or no tessellation at all (Murrieta-Cid et al, 2002; Bandyopadhyay and Ang, 2006; Karaman and Frazzoli, 2011). This implies that obstacles to motions and observations are the same, and also that frequent realistic situations like areas where AGVs can observe through but cannot traverse (*e.g.* ground holes or water ponds), or where AAVs can fly over but not observe under (*e.g.* undergrowth) can not be represented.

Along with the growing computation power, more complex models (2.5D (Kolling et al, 2010), 3D (Bandyopadhyay et al, 2011), or multilayered models (Flushing et al, 2012; Robin and Lacroix, 2012)) have recently been proposed. Such models embody more information and are hence more realistic, allowing finer strategies. For instance, one is able to take advantage of higher vantage points to observe a larger viewshed (Kolling et al, 2010) or to distinguish areas that block motions from areas that block observations (Robin and Lacroix, 2012).

One may also use the now widely available maps to improve the models. For instance, Flushing *et al.* perform *Wilderness Search And Rescue* (WiSAR) with heterogeneous agents with a wide diversity of sensory-motor and cognitive skills (like robots, humans, pets, *inter alia*) (Flushing et al, 2012). The approach relies on a *Geographic Information System*, that provides models with a reasonable accuracy, and manages the emergency response considering the strong influence of terrain characteristics and the environmental conditions on the performances of the various agents.

Besides these more realistic models, some authors exploit highly abstracted representations, like CAT(K) (Alexander et al, 2006, 2010). This often eases the finding of solutions, and allows to rigorously assess algorithmic complexity. However, the loss of information induced by the abstraction may impede the validity of the solutions when confronted to the real world.

b) Discrete Worlds:: The discrete grid or tessellation models are widely used, as they straightforwardly define graphs, upon which algorithms exploiting graph theory can be built (Parsons, 1978; Megiddo et al, 1988; Derenick et al, 2009; Pasqualetti et al, 2010; Murrieta-Cid et al, 2007). However, it is difficult to transform a metrical model of the world into a meaningful topological model (Moors et al, 2005), and only few authors provide means to do so (Hollinger et al, 2010; Kolling and Carpin, 2008; Liu et al, 2015): some directly assume that the graph is available, either handmade, or built by a prior process (Hollinger et al, 2009), others use random sampling as a compromise between the continuous world and a discrete model.

Discretization is also a workaround for the computational complexity of continuous numerical models, even when the algorithms are theoretically valid with continuous models (Soltero et al, 2013). As stated by Bhattacharya "While discretization invariably implies a certain level of approximation and deviation from the original metric space, in order to make any continuous problem computationally feasible it is an indispensable trade-off." (Bhattacharya et al, 2013). As a matter of fact, continuous models are mostly used for local reactive control or greedy decisions (Murrieta-Cid et al, 2002; Sarmiento et al, 2009; Bandyopadhyay and Ang, 2006). On the contrary, discrete models are often used for mathematical proofs in abstract representations (Parsons, 1978; Noori and Isler, 2014; Alexander et al, 2006).

Besides, the assumption of continuity is appealing and eases guarantees and proofs, but it raises an issue: indeed there are discontinuities in real world that impact both motions and visibility and that continuous abstract models may not handle (Israel et al, 2012).

c) Space-time manifold:: Besides the space representation, time representation is crucial, and can actually hardly be decorrelated from the space representation, especially in discretized models. Note that discrete time can be used with continuous space models (Bandyopadhyay and Ang, 2006), but continuous time representations are also used (Alexander et al, 2006). Time discretization is typically used to define a countable number of states, and as highlighted by LaValle (LaValle et al, 1997), "the next state x_{k+1} will usually not lie exactly at a discretized value" (be it temporal or spatial). This means that discretization will probably reduce the coherence between the real state and the modelled state, and one has to ensure that the algorithms are robust to such incoherences.

However, to ease the mathematical proofs, the "one space unit travelled per one time unit" hypothesis is convenient, and often does not lead to any loss of generality (Soltero et al, 2013). But it may impact the quality of the resulting solutions. As a matter of fact, the relation between space and time is really sensitive when accurate coordination is required or assumed, or when a robot cannot "wait" for the others, for instance when pursuing a target (Robin and Lacroix, 2012; Strom et al, 2010). In both cases, experiments will validate or not the assumptions about time, and therefore one has to take great care about how this validation step is led (see Section V-C).

d) Environmental constraints:: The missions considered in our survey can take place in particular environments that challenge the algorithms: some environments can indeed compromise some assumptions, and constrain or even prevent some actions. For instance, some algorithms take for granted that one can mark the environment in a way or an other (Chu et al, 2007; Glad et al, 2010). In this case, one has to pay attention to the feasibility of such markings – and to their cost.

The recent advent of RFIDs offers an interesting technical solution to mark the environment (Mamei and Zambonelli, 2005), but they can hardly be used in every context – a difficulty overlooked by most simulators where marking can be easily emulated. As asserted by Glad *et al.*, "Robotic systems come with their own hypotheses that are more restrictive than in simulation." Here again only realistic experiments should validate the approach (see also Section V-C). For instance, Pugh and Martiloni studied the transformation of a swarm algorithm into a multi-robot search strategy and the impact of the RFID detection distance (which emulates the marking) (Pugh and Martinoli, 2007). In this case, it appears that the performances were not too degraded in small environments.

While most reviewed work tackle simple environments, *i.e.* 2D grounds or volumes with only no fly zones for AAVs, some authors tackle more challenging environments. Various authors address target detection and tracking problems in maritime environments (Yau and Chung, 2012; Svec et al, 2014; Raboin et al, 2013; Kuhlman et al, 2014; Ehlers, 2010). Depending on the considered models, the environment may either slightly impact the approaches or, on the opposite, drive the approach. For instance, Ehlers perform *probabilistic search* with multi-static sonar (Ehlers, 2010) (which have different properties than classic light-based sensors) while Yau and Chung use advanced ocean models that take into account the surface current flows to drive the *probabilistic search* of drifting objects (Yau and Chung, 2012).

2) The agents: The term "agent" stands here for the robots and the targets. For the scenarios considered in this survey, the considered actions are the motions, the observations, and the communications. A model of each of these capacities is required, and its choice is obviously strongly related to the choice of the environment model, as both are combined to assess the outcome of the actions.

a) Motion model:: The model accuracy of the motion capacities, and especially the kinematics and dynamic constraints, varies a lot. The more abstract the world model is, the more the motion model is. With graphs, the motion model only describes if a given node (area) can be reached by the agent or not, possibly with a cost encoded in edge weights (*e.g.* distance, time or energy). It is largely assumed that this cost estimate is good, and how the agent should or will effectively move is not of the concern of the motion model in use. This assumption may be presumptuous, and only validation processes can assess its realism.

Nowadays, accurate pattern-based (sampled) motion models are used for ASVs (Svec et al, 2014) and (winged) AAVs (Tang and Ozguner, 2005; Theodorakopoulos, 2009), and more rarely for AGVs (Yu and Beard, 2011), even though non-holonomy is recurrently modelled (Murrieta-Cid et al, 2011).

Information on the environment geometry may also help to refine or constrain the agents motions (Flushing et al, 2012). However, a precise or realistic motion model is not always required, especially for the target. For instance, assuming an infinite speed for the target yields the definition conservative worst-case-guaranteed strategies (Chung et al, 2011).

b) Sensor model:: The sensor models are mainly used for the robots, but may involve the targets as in stealth tracking (Bandyopadhyay et al, 2005) or when the authors consider the evader point of view (Markov and Carpin, 2007). Sensor models may be very basic: most often only distance matters, especially when the world is modelled as a graph. However, some authors explicitly consider the fields of view and the related visibility constraints (Durham et al, 2011; Gerkey et al, 2006; Murrieta-Cid et al, 2002; Bhattacharya et al, 2007; Vo and Lien, 2010), taking into account the orientation, a maximum angle and a distance. Finer models may also exploit 2.5D or 3D information (Kolling et al, 2010; Bandyopadhyay et al, 2011), or a multi-layered world (Robin and Lacroix, 2012). Most considered sensors are light-based sensors, be it cameras or LIDARs (except for AUVs, where the sensor of choice is sonar (Ehlers, 2010)), and so most of visibility constraints are line-of-sight. Taking into account missed detections and false-positives also leads to interesting considerations on uncertainties (Raboin et al, 2013).

c) Communications model:: Although the robots' communication abilities may be intuitively considered as very similar to sensing (when there is a visibility link, one may reasonably assume that there is a communication link), most of the authors use the full connectivity assumption. However convenient this assumption is, its realism is questionable and may strongly impact the efficiency of the approach. Indeed, accurate communication models are complex and expensive (Mosteo, 2010). This explains why some authors define approaches upon the communication requirements and issues (Hollinger and Singh, 2012), or specifically study its impact (Pugh and Martinoli, 2007; Hollinger and Singh, 2010).

d) Expected behaviour:: Besides the models of actions (motion, sensing, communicating), one may want to model the expected behaviour of the target or of the other teammates. The target model is often a probabilist motion model, be it random walk (Parker, 2002; Hollinger and Singh, 2010; Strom et al, 2010) or more elaborate like Bayesian or Markovian models (Yu and Beard, 2011; Hollinger et al, 2009; Bandyopadhyay et al, 2009). Predicting the target's behaviour and motion allows elaborating more sophisticated or less conservative strategies. In the adversarial case for instance, the target behaviour can be accurately considered using a game theoretic approach (Murrieta-Cid et al, 2007, 2008; Vieira et al, 2009).

Various target models are also used as a metric to compare algorithms (Sak et al, 2008). Indeed the model of the target's behaviour does matter, as finer strategies are possible when the model is correct, but they can turn to be counter-productive when the model is not adequate (Yu and Beard, 2011).

Modelling the teammates behaviour is also relevant: the main motivation is to reduce the need for communication, as implicit coordination built on the prediction of others motions can be achieved (Grocholsky et al, 2005). The models may be

hard coded (Bayesian, POMDP) or learned (Xu et al, 2013).

B. Approaches

Robotics being at the crossroads of numerous disciplines, the state of the art contains a large variety of approaches tackling similar – if not identical – problems. Among the numerous criteria that define an approach, we distinguish (i) the centralized systems from the decentralized ones, (ii) the cooperative patterns from the "selfish" ones, considering both implicit and explicit cooperation, (iii) the consideration of uncertainties, especially through probabilistic models, and (iv) the planning processes from the optimization processes. Note that the selection of an approach is obviously associated to the definition of the chosen models.

1) Centralized and decentralized systems: The problems considered in this survey are mainly multi-robot problems, hence the question of centralized and decentralized systems arises. Numerous centralized algorithms have been proposed until the early 00's but also more recently (Ohsumi, 1991; Hespanha et al, 1999; González-Baños and Latombe, 2001; Tang and Ozguner, 2005; Ruan et al, 2005). Centralized systems are more convenient to provide global optimality. However, they face strong constraints in the real world. Indeed, they often require full connectivity, are sensitive to dynamic environments, especially when solutions are computed off-line, and most of them hardly scale up, in particular with the number of robots (Moors et al, 2005).

Most target related problems are at least NP-hard, and as noticed in Section III-C, one cannot expect high performances with centralized algorithms. One should instead focus on the benefits brought by decentralized systems (Hollinger and Singh, 2010). The latter are more robust, generally scale well with the number of robots, and are more adapted to real world constraints (dynamic environment, communication constraints, *inter alia*), easing the integration process (Moseley et al, 2009; Jung and Sukhatme, 2002; Renzaglia, 2012; Kalra et al, 2005). Decentralized solutions generally provide suboptimal solutions with only local optimality (Cao et al, 2006; Casbeer and Kingston, 2006; Chu et al, 2007). Still, they provide interesting performances, especially under realistic constraints, and constitute the current main research trend (Vidal et al, 2002; Parker and Emmons, 1997; Zhou et al, 2004; Mottaghi and Vaughan, 2007; Chu et al, 2007; Kolling and Carpin, 2010a; Ataei et al, 2013).

2) The need for cooperation: Most of the surveyed problems require several robots to be solved adequately, and the quality of the solutions is generally improved as a team of robots allows more flexibility in the strategies.

There are different ways to organize robot teams. The robots may cooperate or simply perform their tasks independently, following a prior task allocation. The latter case is well illustrated by both the cycle- and partition-based patrolling schemes (Section III-D), whose main advantage is that they do not require any communication, avoiding the associated issues. The drawback is the weaker robustness to robot failures or changes in the environment or the mission definition, as robots cannot modify the team strategy globally.

On the other end, on-line cooperation helps to solve problems efficiently, and is even required for the problems which involve tight coordination (*capture*), or dynamic adaptation to incoming data such as target observation (*observation*, *target localization*, *probabilistic search*). The benefits of coordination are tightly related to the structure of the environment: highly structured worlds like office environment benefit less from a tight coordination than open environments (Amigoni et al, 2013).

The cooperation can either be implicit or explicit. The latter allows to finely control the resulting system, because every decision and action are discussed and broadcasted through the whole team (or at least the surroundings agents). However, there are issues with the communication load and the combinatorial complexity of the decision processes.

Implicit cooperation often rely on cooperative control (Murray, 2007), while explicit cooperation is often achieved through *task allocation*, where the goal is to allocate a set of tasks between several agents to optimize various criteria. The large literature about task allocation is beyond the scope of this survey, but it is worth noticing that the target detection problem has often been used to compare or benchmark allocation algorithms. According to our taxonomy, they mainly tackle the *hunting* problem as they typically provide no guarantee, their goal being to allocate a set of task and not to perform efficient search – even though the tasks are ranked or prioritized.

Task allocation processes can be solved in a fully distributed way and have been integrated on real robots. If the implemented strategies are not impressive (they most often are parallel randomized search), they are quite robust to drifts, lack of information or communication issues. One may refer to Jennings' MOVER system (Jennings et al, 1997) (fully parallel and distributed cooperation based on the integration of single agents behaviour), to Pirjanian using multi-objective cooperation (with Paretto optimality for the decision process) (Pirjanian and Mataric, 2000), or to Cai for an auction approach (Cai et al, 2008) which performs dynamic coalition formation to transform a *Multiple Pursuer Multiple Evader Game* problem to several *Multiple Pursuer Single Evader Games* problems, easier to tackle. Raboin's usage of market-based mechanism is also interesting because the considered ASVs can perform multitasking and because some task require several robots (Raboin et al, 2013) – it is a MT-MR-TA problem (Multi-Task robots, Multi-Robot tasks, Time-extended Allocation) variant of the task allocation problems according to Gerkey's taxonomy (Gerkey and Mataric, 2004).

3) Uncertain and dynamic environments: Inherently, robotic systems face many uncertainties: their sensors are prone to a variety of errors and noise, the agents behaviour can hardly be predicted accurately as results of actions are uncertain, communications fail, *inter alia*. Targets identification may also be subject to uncertainty (neutral or foe?) (Raboin et al, 2013). Ignoring these uncertainties will likely result in a defective system when facing the real world. Two main strategies, which can be coupled, help to prevent this: taking into account uncertainties at the modelling level, and replanning on-line when the gap between the modelled state and the real state is too large.

Probabilistic models have gained interested over the past twenty years in all robotic problems, including the ones considered here. This is explained by the need to improve the models on which planning rely and by the advent of computing performances which allows to handle these models. The trendy probabilistic models are, in a rough chronological order, classical probabilities (Hespanha et al, 1999; Vidal et al, 2002; El-Rayes et al, 2003; Moors et al, 2005; Sarmiento et al, 2009; Strom et al, 2010; Gan et al, 2012; Raboin et al, 2013), Bayesian models (Wong et al, 2005; Mottaghi and Vaughan, 2007; Santos and Lima, 2010; Portugal and Rocha, 2013; Qian et al, 2008; Pita et al, 2009), Markovian models (MDP) (Agmon et al, 2008; Ruan et al, 2005; Ohsumi, 1991; Amigoni et al, 2009), and partially observed Markovian models (POMDP) (Ferrari et al, 2009; Ehlers, 2010; Hollinger et al, 2009; Bandyopadhyay et al, 2009). The latter are the more appealing but their extensive computational cost prevents their application to large instances of problems (Roy et al, 2005; Bandyopadhyay et al, 2009). Probabilistic solutions tend to yield finer strategies but at a significant computational cost, and most importantly they remain sensitive to the modelling phase. For instance, determining the best order of Markovian models has a strong influence on performance, but is not trivial to achieve (Yu and Beard, 2011).

Numerous problems tackled here are NP-hard at least, and the computational cost is heavy with probabilistic models, which explains why many solutions are computed off-line for problem instances of reasonable size (Bhattacharya et al, 2013; Kolling and Carpin, 2010b; Alamdari et al, 2013; Moors et al, 2005). However, the real world is rarely static, and one often has to react to external events (like target motion) without being able to precompute strategies for each and every possible state (the state space being globally intractable). In such cases on-line computing is required. It brings robustness as one is able to recompute a valid solution on demand, when needed. However, on-line computing often comes with local considerations and thus suboptimality. This dilemma is well illustrated in Section IV-B: the *following* problem has been shown to be entirely decidable, but is NP-complete (Murrieta-Cid et al, 2008). This is why the efficient state-of-the-art solutions consider only local information and are computed on-line (Murrieta-Cid et al, 2002; Bandyopadhyay and Ang, 2006). They are not optimal, but perform well, even in difficult environments.

4) Planning vs. Optimization: We surveyed a variety of detection and tracking problems that roughly come to determine *who does what, when, and where*. Most are formulated either as *planning* problems or as *optimization* problems. While the underlying problem remains the same, the formulation differs a lot, and so do the solutions and the results.

Planning comes to simulate actions and their effects in order to decide the actions sequence to perform to reach a given state. Several formalisms exist to implement planners, and the critical stage is the modelling of the action space, the action effects and the world states. One of the many available planners will provide a solution on the basis of these models.

In the formalism of *optimization*, an objective function to optimize with respect to some constraints is defined. More than the choice of a solver, the critical stages are the quantification of the world states and actions, and the definition of the objective function.

One key difference is that optimization requires numerical values, while traditional planning work with qualitative statements. Correctly evaluating these numerical values may be straightforward (*e.g.* when trying to minimise the *idleness* in *patrolling*), but also more complex for others (like *capture* for instance). Optimization is close to raw data and commands while traditional planning is semantically expressive and allows a higher level of abstraction in the models, easing hierarchical planning (*e.g.* the *Hierarchical Task Networks* (Erol, 1996)). Besides, a hierarchical decomposition allows to mix both traditional planning (*e.g.* to roughly define tasks) and optimization (to refine the local solution). Recently, thanks to the increase of computing power, stochastic optimization has proven to be efficient at solving complex problems.

C. Results analysis and validation process

Among every problem analysed in this survey, one can distinguish two trends: a large part of the contributions focus on experimental results (Ataei et al, 2013; Kalra et al, 2005; Bandyopadhyay et al, 2005; Strom et al, 2010), while others provide theoretical results (Agmon et al, 2008; Alexander et al, 2010; Elmaliach et al, 2009; LaPaugh, 1982; Eaton and Zadeh, 1962) – this is a coarse partition, and some papers propose both kinds of results (Zhou and Roumeliotis, 2011; Hollinger and Singh, 2012; Alamdari et al, 2013).

Theoretical results are essential as they light the way to efficient and practical solutions. Numerous theoretical results state the complexity of a given class of problems (Hollinger and Singh, 2012; O'rourke, 1987; Megiddo et al, 1988; Kolling and Carpin, 2010b; Isler et al, 2005; Hollinger et al, 2009): in case of NP-difficult problems, one should probably focus on off-line computation, or suboptimal on-line solutions. Some papers also give hints on the solutions: for instance, LaPaugh proved that "recontamination" does not help for solving the *capture* problem (LaPaugh, 1982). But whatever the theoretical guarantees on complexity or optimality, experiments are required to assess the applicability of the proposed algorithms. For instance, Eaton and Zadeh "solved" the *following* problem in 1962 (Eaton and Zadeh, 1962), but the proposed solution is actually not applicable, and to our knowledge has not been implemented on-board any robot: it operates in an abstract space which is not correlated to the real world – or at least no one provided a valid transformation between the real world and this abstract space. There have been much work on *following* since then: several approaches show good performances, but they are far from Zadeh's solution. Furthermore, most analytical approaches are not applicable in real conditions considering the computational constraints.

This illustrates that algorithms can only be validated with integration: only testing them on-board robots in an actual context will assess that they can cope with the challenging issues that are uncertainties, non-determinism, asynchronicities, *etc.* Not considering these common issues yields solutions that are not robust. As stated by Parker about the *observation* problem:

"analytical techniques have been developed for solving this problem in complex geometrical environments. However, these previous approaches are very computationally expensive – at least exponential in the number of robots – and cannot be implemented on robots operating in real-time" (Parker, 2002). Similarly, Portugal and Rocha state that "it is the authors belief that research in this field should be more oriented towards effective solutions with applicability in the real world" (Portugal and Rocha, 2013). These statements are in defense of experimental results, *i.e.* "solutions that works for real".

Although the approaches focusing primarily on integration and experiments can be criticized, we stand for the importance of such work, that addresses what matters in the end, that is solutions that effectively solve real world problems. These approaches often rely on local considerations, and result in efficient and sometimes elegant systems (Bandyopadhyay et al, 2005; Murrieta-Cid et al, 2002; Grocholsky et al, 2005).

Yet, most papers lack actual validation. Of course authors usually try to provide a fair description of their experimental protocol (parameters, number of runs, description of the simulator, *etc.*) but very few present realistic testing environments. The proposed approaches are mostly validated with simplified simulations (Tang and Ozguner, 2005; Sarmiento et al, 2009; Kolling et al, 2010; Durham et al, 2011; Isler et al, 2005; Miao et al, 2010; Zhou et al, 2004; Chu et al, 2007; Vo and Lien, 2010; Amigoni et al, 2009), *i.e.* simulations that neglect to represent most of real world characteristics. These simulated worlds are indeed often discretized in time and space by construction, and there are no or few uncertainties. In this context, the initial assumptions and simplifications required by the algorithms are introduced into the simulators, which can then hardly exhibit the solution flaws. Experiments on-board robots may present the same limits when they twist the reality to fit the models (*e.g.* restraining the agents movements to an artificial grid-based space (Yu and Beard, 2011; Portugal and Rocha, 2013) or artificially enforcing the time discretization (Strom et al, 2010)). We do understand many of these simulations were constructed carefully to demonstrate the proposed approach while abstracting away things like vehicle dynamics that are not necessarily relevant to the problem at hand. Nonetheless, in practice such simulations tend to hide or neglect important aspects that may question the validity of the evaluated approaches. This gap between models and reality have been spotted and studied by several authors (Jung and Sukhatme, 2002; Pimenta et al, 2010; Glad et al, 2010; Portugal and Rocha, 2013).

The validation process often present other flaws as the lack of state-of-the-art comparisons, algorithms being often "benchmarked" against trivial solutions like random walkers or brute-force solutions (Isler et al, 2005; Portugal and Rocha, 2010; Miao et al, 2010; Israel et al, 2012), or the lack of "culture of statistics", results being rarely statistically significant. Some of these flaws are understandable: thorough validations call for a lot of engineering, numerous tests and a logistic out of reach of most researchers. Also, fair comparisons are restrained by the variety of robots and testing environments between teams and labs.

Nevertheless, some authors provide nice state-of-the-art comparisons (Zhou et al, 2004; Chu et al, 2007; Markov and Carpin, 2007; Vo and Lien, 2010) and others have recently introduced good statistic analysis (Sak et al, 2008; Amigoni et al, 2013), computing for instance standard deviations and p-values when it is relevant. Efforts are also made to homogenize the testing platforms or to reuse past testing environments or simulators to fairly compare algorithms (Strom et al, 2010). Challenges like RoboCup (including the Rescue Virtual Robot Competition (Balakirsky, 2007; Jacoff et al, 2003)) are also a way to compare the performance more fairly, and some papers present benchmarks of various algorithms and compare them through a set of metrics (Jacoff et al, 2003; Sak et al, 2008), which offer references in the considered domains.

D. Open problems

Following our synthetic and transverse analysis presented in the previous sections, we have identified what we consider as the three most important areas and directions on which the research communities should maintain or pursue their efforts: (i) improving the models, (ii) developing decentralized algorithms, and (iii) strengthening the validation processes. Those problems are major open problems that go beyond communities or specific problem formulations, and we do not aim at listing all the specific open variations of the tackled problems; indeed, more precise open problems would not be relevant with respect to the spirit of our survey, and the reader interested by this kind of results should instead refer to the partial conclusion in the paper body or to the cited area-specific surveys.

1) Towards more expressive models: These last years, finer models have been proposed: they embrace probabilistic considerations, multi-layered world models and hierarchical representations. These improved models are mostly tractable now thanks to advances in sensing and increasing computational power. Through their extended expressiveness, they meet the need in robotics to fill, or at least to reduce the gap between models and reality. The first results given by these finer models are promising, and one may wish and expect that they become the new state-of-the-art standard, replacing the common but limited 2D mono-layer maps. Communications remain however to be modelled (and considered) in a more realistic manner. This may imply direct links, but also the consideration of quality network maps embodying the recent advances of 3G/4G/LTE/Satellite data networks. The benefits and drawbacks of more realistic models should also be more extensively studied in order to point out the possible trade-offs.

2) Towards decentralized systems: Considering the number of problems analysed in our survey that are NP-hard at least, we think that suboptimal yet efficient real-time algorithms are the key to solve the real cases scenarios while handling the dynamics changes on-line. There have been an increasing number of pieces of work on decentralized algorithms, most of them showing descent scalability. Yet, many decentralized algorithms are derived versions of a prior centralized approach. Although this is a comprehensive approach, it remains limited, and decentralized algorithms designed "from scratch" may lead to better cooperative schemes. Furthermore, there has

been some interesting work on "where and when should we cooperate", pointing out several situations and contexts where the benefits of explicit cooperation are limited. This could be further studied, along with the impact on the communication load. Besides, anytime algorithms may also be a bridge between efficiency and optimality when facing complex problems (Hollinger et al, 2010), especially when time matters as in *search and rescue* scenarios. Finally, most of the time only experimental bounds on the performances are given (this is particularly true for decentralized algorithms), and we yearn to see more theoretical bounds on the suboptimality of the resulting solutions.

3) Towards better validation processes: In our analysis, we pointed out several recurrent flaws in the validation processes. Most can be fixed at a small cost, and there should be drastic changes in the validation processes of the proposed approaches.

Too many papers provide unrealistic test conditions which prevent the evaluation of such applicability. Rigorous validations should involve reproducible results, with a statistical analysis and a fair comparison to other state-of-the-art solutions. We believe that realistic simulations is a key to tackle the validation issues, as they provide common testing environments with realistic conditions, while easing statistically significant experiments. Realistic simulators differ from simplified ones in the way they model reality: the simulated space and time are continuous, and they generally embed a physical engine. Their development requires engineering efforts, but fortunately one can now easily find off-the-shelf open source simulators (*e.g.* Morse (Echeverria et al, 2012), Gazebo (Koenig and Howard, 2004), or USARSim (Balakirsky, 2007)), and efforts are currently put on the development of more realistic simulators (R4SIM, 2015). These simulators come with various robot models and environments, and using a common framework and a modular architecture like ROS (Quigley et al, 2009) would even allow to compare strategies against one another (*e.g.* a pursuer strategy against an evader strategy elaborated by another team) and to benefit from work in other sub-domains of robotics (*e.g.* robot localization or sensing).

This is the continuation of a hierarchical approach, where basic algorithms are validated in high-level simulations and then additional low-level issues are worked in higher-fidelity simulations. In an ideal realistic modular approach, high-level and low-level algorithms would be integrated in a same framework to properly show their consistency with each other and their adequacy with the reality and its common issues. Hence, validations that involve open source codes, datasets and scenarios defined in realistic simulators can provide both reproducible experiments and statistically significant results and comparisons, and would also pool efforts in the communities, and ease the adoption of breakthroughs from related communities. Making the code publicly available to foster experimental reproducibility and comparison is definitely a good practice for the validation process ((Amigoni and Basilico, 2010), among others), which should naturally be encouraged. Providing such a framework and datasets is still a long way to go but many tools are already around to make it feasible in a nearby future.

Finally, the ultimate validation is implementation and tests

in real systems – and in realistic conditions, not laboratory engineered environments. For this purpose, challenges and competitions[10] are ideal to foster the development of robust realistic systems.

VI. CONCLUSION

This paper presents a unifying and clarifying taxonomy of the various target related robotics decisional problems. It defines classes of missions and problems that go beyond the frontiers of specific communities and specific problems. Through this agnosticism, it gathers and reviews comparable work that would otherwise be put apart due to distinct communities and associated specific key works: it thus enlarges the scope of prior surveys.

Following the taxonomy and the survey, the paper analyses the transverse models and approaches that are recurrent through all the tackled problems. From this overall analysis, we highlight three main current and general open problems, which are general directions for which we think the community should pursue or increase its efforts to gather new results: (i) to propose and to exploit better and richer models, which we think is the primal concern, (ii) to design intrinsically decentralized algorithms, and (iii) to strengthen the validation process.

Acknowledgements: The authors are thankful to Wheeler Ruml for his helpful remarks.

REFERENCES

Acevedo JJ, Arrue BC, Maza I, Ollero A (2014) A decentralized algorithm for area surveillance missions using a team of aerial robots with different sensing capabilities. In: IEEE International Conference on Robotics and Automation (ICRA), pp 4735–4740

Agmon N, Kraus S, Kaminka GA (2008) Multi-robot perimeter patrol in adversarial settings. In: IEEE International Conference on Robotics and Automation (ICRA), pp 2339–2345, DOI 10.1109/ROBOT.2008.4543563

Al Marzouqi M, Jarvis RA (2011) Robotic covert path planning: A survey. 2011 IEEE 5th International Conference on Robotics, Automation and Mechatronics (RAM) pp 77–82, DOI 10.1109/RAMECH.2011.6070460

Alamdari S, Fata E, Smith SL (2013) Persistent monitoring in discrete environments: Minimizing the maximum weighted latency between observations. International Journal of Robotics Research (IJRR) DOI 10.1177/0278364913504011

Alexander S, Bishop R, Ghrist R (2006) Pursuit and evasion in non-convex domains of arbitrary dimensions. In: Proceedings of Robotics: Science and Systems (RSS)

Alexander S, Bishop R, Ghrist R (2009) Capture pursuit games on unbounded domains. LEnseignement Mathématique 55 pp 103–125

Alexander S, Bishop R, Ghrist R (2010) Total curvature and simple pursuit on domains of curvature bounded above. Geometriae Dedicata pp 1–15, arXiv:0909.4113v1

---

[10]such as the Multi-Autonomous Ground Robot International Challenge (MAGIC) held in 2010 (Hsieh and Lacroix, 2012) or the Eurathlon series of challenges

Almeida A, Ramalho G, Santana H, Corruble V, Chevaleyre Y (2004) Recent Advances on Multi-Agent Patrolling. Advances in Artificial Intelligence

Alspach B (2006) Searching and sweeping graphs: a brief survey. Le matematiche LIX:5–37

Amigoni F, Basilico N (2010) A Decision-Theoretic Framework to Select Effective Observation Locations in Robotic Search and Rescue Scenarios. In: IEEE International Conference on Robotics and Automation (ICRA)

Amigoni F, Basilico N, Gatti N (2009) Finding the Optimal Strategies for Robotic Patrolling with Adversaries in Topologically-Represented Environments. In: IEEE International Conference on Robotics and Automation (ICRA)

Amigoni F, Basilico N, Li A (2013) How much worth is coordination of mobile robots for exploration in search and rescue? RoboCup 2012: Robot Soccer World Cup XVI

Annas J, Xiao J (2009) Intelligent Pursuit & Evasion in an Unknown Environment. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp 4899–4906

Annas J, Xiao J (2010) Intelligent Pursuit & Evasion in Unknown Environments Against Human Players - extended abstract. In: IEEE International Conference on Robotics and Automation (ICRA)

Ataei HN, Ziarati K, Eghtesad M (2013) A BSO-Based Algorithm for Multi-robot and Multi-target Search. Recent Trends in Applied Artificial Intelligence pp 312–321

Balakirsky S (2007) Usarsim: a robocup virtual urban search and rescue competition. Defense and Security Symposium

Bandyopadhyay T, Ang M (2006) A greedy strategy for tracking a locally predictable target among obstacles. IEEE International Conference on Robotics and Automation (ICRA) pp 2342–2347, DOI 10.1109/ROBOT.2006.1642052

Bandyopadhyay T, Ang Jr MH, Hsu D (2005) Stealth Tracking of an Unpredictable Target among Obstacles. Algorithmic Foundations of Robotics VI pp 43–58

Bandyopadhyay T, Rong N, Ang M, Hsu D, Lee WS (2009) Motion Planning for People Tracking in Uncertain and Dynamic Environments. In: IEEE International Conference on Robotics and Automation (ICRA)

Bandyopadhyay T, Jr MA, Hsu D (2011) Motion planning for 3D target tracking among obstacles. Robotics Research

Benkoski SJ, Monticino MG, Weisinger JR (1991) A survey of the search theory literature. Naval Research Logistics (NRL) 38(4):469–494

Bhattacharya S, Hutchinson S (2009) On the Existence of Nash Equilibrium for a Two-player Pursuit–Evasion Game with Visibility Constraints. International Journal of Robotics Research (IJRR) 29(7):831–839, DOI 10.1177/0278364909354628

Bhattacharya S, Candido S, Hutchinson S (2007) Motion Strategies for Surveillance. In: Proceedings of Robotics: Science and Systems (RSS)

Bhattacharya S, Ghrist R, Kumar V (2013) Multi-robot coverage and exploration on Riemannian manifolds with boundaries. International Journal of Robotics Research (IJRR) 33(1):113–137, DOI 10.1177/0278364913507324

Bopardikar SD, Bullo F, Hespanha JP (2007) Cooperative pursuit with sensing limitations. In: American Control Conference (ACC), pp 5394–5399, DOI 10.1109/ACC.2007.4282474

Borie R, Tovey C, Koenig S (2011) Algorithms and complexity results for graph-based pursuit evasion. Autonomous Robots 31(4):317–332, DOI 10.1007/s10514-011-9255-y

Cai Z, Sun L, Gao H, Zhou P (2008) Multi-robot Cooperative Pursuit Based on Task Bundle Auctions. Intelligent Robotics and Applications pp 235–244

Cao Z, Tan M, Li L, Gu N, Wang S (2006) Cooperative Hunting by Distributed Mobile Robots Based on Local Interaction. IEEE Transactions on Robotics 22(2):403–407

Casbeer D, Kingston D (2006) Cooperative forest fire surveillance using a team of small unmanned air vehicles. International Journal of Systems Science 37(6)

Choset H (2000) Coverage of Known Spaces: The Boustrophedon Cellular Decomposition. Autonomous Robots 9(3):247–253, DOI 10.1023/A:1008958800904

Choset H (2001) Coverage for robotics A survey of recent results. Annals of Mathematics and Artificial Intelligence 31(1-4):113–126, DOI 10.1023/A:1016639210559

Chu HN, Glad A, Simonin O, Sempe F, Drogoul A, Charpillet F (2007) Swarm approaches for the patrolling problem, information propagation vs. pheromone evaporation. In: IEEE International Conference on Tools with Artificial Intelligence (ICTAI), pp 442–449

Chung TH, Hollinger Ga, Isler V (2011) Search and pursuit-evasion in mobile robotics. Autonomous Robots 31(4):299–316, DOI 10.1007/s10514-011-9241-4

Clark J, Fierro R (2007) Mobile robotic sensors for perimeter detection and tracking. ISA transactions 46(1):3–13, DOI 10.1016/j.isatra.2006.08.001

Cole DT, Thompson P, Göktogan AH, Sukkarieh S (2010) System development and demonstration of a cooperative UAV team for mapping and tracking. International Journal of Robotics Research (IJRR) 29(11):1371–1399, DOI 10.1177/0278364910364685

Cowley A, Hsu HC, Taylor CJ (2004) Distributed sensor databases for multi-robot teams. In: IEEE International Conference on Robotics and Automation (ICRA)

Daniel K, Borie R, Koenig S, Tovey C (2010) ESP: Pursuit Evasion on Series-Parallel Graphs (Extended Abstract). In: International Conference on Autonomous Agents and Multiagent Systems (AAMAS), pp 4–5

Derenick J, Spletzer J, Hsieh A (2009) An Optimal Approach to Collaborative Target Tracking with Performance Guarantees. Journal of Intelligent and Robotic Systems 56(1-2):47–67, DOI 10.1007/s10846-008-9302-x

Dornhege C, Kleiner A, Kolling A (2013) Coverage Search in 3D. In: IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), pp 1–8, DOI 10.1109/SSRR.2013.6719340

Durham JW, Franchi A, Bullo F (2011) Distributed Pursuit-Evasion without Mapping or Global Localization via Local Frontiers. Autonomous Robots

Eaton JH, Zadeh L (1962) Optimal pursuit strategies in discrete-state probabilistic systems. Journal of Fluids Engineering 84(1):23–29

Echeverria G, Lemaignan S, Degroote A, Lacroix S, Karg M,

Koch P, Lesire C, Stinckwich S (2012) Simulating Complex Robotic Scenarios with MORSE. In: Simulation, Modeling, and Programming for Autonomous Robots (SIMPAR), Springer, pp 197–208

Ehlers F (2010) Multi-Robot Teamwork in Multistatic Sonar. In: ICRA Workshop: Search and Pursuit/Evasion in the Physical World

El-Rayes AB, Mohamed AEMA, Gabal HMA (2003) Linear Search for a Brownian target motion. Acta Mathematica Scientia 23(3):321–327

Elmaliach Y, Agmon N, Kaminka GA (2009) Multi-Robot Area Patrol under Frequency Constraints. Annals of Mathematics and Artificial Intelligence 57:293–320, DOI 10.1007/s10472-010-9193-y

Erol K (1996) Hierarchical Task Network Planning: Formalization, Analysis, and Implementation. PhD thesis, University of Maryland

Fazli P, Davoodi A, Pasquier P, Mackworth AK (2010) Fault-Tolerant Multi-Robot Area Coverage with Limited Visibility. In: ICRA Workshop: Search and Pursuit/Evasion in the Physical World

Ferrari S, Fierro R, Tolic D (2009) A Geometric Optimization Approach to Tracking Maneuvering Targets Using a Heterogeneous Mobile Sensor Network. In: IEEE Conference on Decision and Control, held jointly with the Chinese Control Conference (CDC-CCC), pp 1080–1087

Flushing EF, Gambardella L, Caro GAD (2012) GIS-based Mission Support System for Wilderness Search and Rescue with Heterogeneous Agents. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Workshop on Robots and Sensors integration in future rescue INformation system (ROSIN)

Gan SK, Fitch R, Sukkarieh S (2012) Real-time decentralized search with inter-agent collision avoidance. In: IEEE International Conference on Robotics and Automation (ICRA), pp 504–510, DOI 10.1109/ICRA.2012.6224975

Gerkey BP, Matarić MJ (2004) A formal analysis and taxonomy of task allocation in multi-robot systems. International Journal of Robotics Research (IJRR) 23(9):939–954

Gerkey BP, Thrun S, Gordon G (2006) Visibility-based pursuit-evasion with limited field of view. International Journal of Robotics Research (IJRR) 1(c):20–27

Ghosh SK (2009) Approximation algorithms for art gallery problems in polygons. In: Conference on Discrete Mathematics

Glad A, Simonin O, Buffet O, Charpillet F (2008) Theoretical Study of Ant-based Algorithms for Multi-Agent Patrolling. Proceedings - 2008 European Conference on Artificial Intelligence

Glad A, Simonin O, Buffet O, Charpillet F (2010) Influence of Different Execution Models on Patrolling Ant Behaviors: from Agents to Robots. In: International Conference on Autonomous Agents and Multiagent Systems (AAMAS)

González-Baños H, Latombe JC (2001) A randomized art-gallery algorithm for sensor placement. In: 7th annual symposium on Computational Geometry

Grocholsky B, Swaminathan R, Keller J, Kumar V, Pappas G (2005) Information Driven Coordinated Air-Ground Proac-tive Sensing. In: IEEE International Conference on Robotics and Automation (ICRA), pp 2223–2228

Hereford J, Siebold M (2010) Bio-inspired search strategies for robot swarms. Swarm Robotics, From Biology to Robotics

Hespanha JP, Kim HJ, Sastry S (1999) Multiple-agent probabilistic pursuit-evasion games. In: IEEE Conference on Decision and Control (CDC)

Hollinger G, Singh S (2010) Multi-robot coordination with periodic connectivity. In: IEEE International Conference on Robotics and Automation (ICRA)

Hollinger G, Singh S, Djugash J, Kehagias a (2009) Efficient Multi-robot Search for a Moving Target. International Journal of Robotics Research (IJRR) 28(2):201–219, DOI 10.1177/0278364908099853

Hollinger G, Kehagias A, Singh S (2010) GSST: anytime guaranteed search. Autonomous Robots 29(1):99–118, DOI 10.1007/s10514-010-9189-9

Hollinger GA, Singh S (2012) Multirobot Coordination With Periodic Connectivity: Theory and Experiments. IEEE Transactions on Robotics 28(4):967–973, DOI 10.1109/TRO.2012.2190178

Hollinger GA, Sukhatme G (2013) Sampling-based motion planning for robotic information gathering. In: Proceedings of Robotics: Science and Systems (RSS)

Hsieh A, Lacroix S (eds) (2012) Special issue on "Multiple Collaborative Field Robots", Journal of Field Robotics, vol 29(5)

Isler V, Khanna S, Spletzer J, Taylor CJ (2005) Target Tracking with Distributed Sensors: The Focus of Attention Problem. Computer Vision and Image Understanding pp 1–37

Israel M, Khmelnitsky E, Kagan E (2012) Search for a mobile target by ground vehicle on a topographic terrain. In: 2012 IEEE 27th Convention of Electrical and Electronics Engineers in Israel, DOI 10.1109/EEEI.2012.6377123

Jacoff A, Messina E, Weiss BA, Tadokoro S, Yuki Nakagawa (2003) Test arenas and performance metrics for urban search and rescue robots. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp 3396–3403

Jennings JS, Whelan G, Evans WF (1997) Cooperative Search and Rescue with a Team of Mobile Robots. In: IEEE International Conference on Advanced Robotics (ICAR), pp 193–200

Jung B, Sukhatme GS (2002) Tracking Targets Using Multiple Robots: The Effect of Environment Occlusion. Autonomous Robots pp 191–205

Kalra N, Ferguson D, Stentz A (2005) Hoplites: A market-based framework for planned tight coordination in multi-robot teams. In: IEEE International Conference on Robotics and Automation (ICRA)

Kamath S, Meisner E, Isler V (2007) Triangulation based multi target tracking with mobile sensor networks. In: IEEE International Conference on Robotics and Automation (ICRA)

Karaman S, Frazzoli E (2011) Incremental sampling-based algorithms for a class of pursuit-evasion games. Algorithmic Foundations of Robotics IX

Karnad N, Isler V (2009) Lion and man game in the presence of a circular obstacle. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp 5045–

5050, DOI 10.1109/IROS.2009.5354443

Katsev M, Yershova A, Tovar B, Ghrist R, LaValle SM (2011) Mapping and Pursuit-Evasion Strategies For a Simple Wall-Following Robot. IEEE Transactions on Robotics 27(1):113–128, DOI 10.1109/TRO.2010.2095570

Katsilieris F, Lindhé M, Dimarogonas D (2010) Demonstration of multi-robot search and secure. In: IEEE International Conference on Robotics and Automation (ICRA)

Kloder S, Hutchinson S (2007) Barrier coverage for variable bounded-range line-of-sight guards. In: IEEE International Conference on Robotics and Automation (ICRA), pp 10–14

Koenig N, Howard A (2004) Design and use paradigms for gazebo, an open-source multi-robot simulator. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp 2149–2154

Kolling A, Carpin S (2007) Cooperative Observation of Multiple Moving Targets: an algorithm and its formalization. International Journal of Robotics Research (IJRR) 26(9):935–953, DOI 10.1177/0278364907080424

Kolling A, Carpin S (2008) Extracting surveillance graphs from robot maps. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp 2323–2328, DOI 10.1109/IROS.2008.4650763

Kolling A, Carpin S (2010a) Multi-robot pursuit-evasion without maps. In: IEEE International Conference on Robotics and Automation (ICRA)

Kolling A, Carpin S (2010b) Pursuit-Evasion on Trees by Robot Teams. IEEE Transactions on Robotics 26(1):32–47, DOI 10.1109/TRO.2009.2035737

Kolling A, Kleiner A, Lewis M, Sycara K (2010) Pursuit-Evasion in 2.5D based on Team-Visibility. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)

Kuhlman M, Svec P, Kaipa K, Sofge D, Gupta S (2014) Physics-aware informative coverage planning for autonomous vehicles. In: IEEE International Conference on Robotics and Automation (ICRA), pp 4741–4746, DOI 10.1109/ICRA.2014.6907553

LaPaugh A (1982) Recontamination does not help. Princeton University, Computer Science Dept

LaValle SM, Gonzales-Banos HH, Becker C, Latombe Jc (1997) Motion Strategies for Maintaining Visibility of a Moving Target. In: IEEE International Conference on Robotics and Automation (ICRA), pp 731–736

Liu M, Colas F, Oth L, Siegwart R (2015) Incremental topological segmentation for semi-structured environments using discretized GVG. Autonomous Robots 38(2):143–160, DOI 10.1007/s10514-014-9398-8

Mamei M, Zambonelli F (2005) Spreading pheromones in everyday environments through RFID technology. In: IEEE Swarm Intelligence Symposium (SIS), pp 281–288

Markov S, Carpin S (2007) A cooperative distributed approach to target motion control in multirobot observation of multiple targets. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)

Megiddo N, HAKIMI SL, Garey MR, Johnson DS, Papadimitriou CH (1988) The Complexity of Searching a Graph. Journal ofthe Association for Computing Machinery

Miao YQ (2010) A study of mobility models in mobile surveillance systems. PhD thesis, University of Waterloo

Miao YQ, Khamis A, Kamel MS (2010) Applying anti-flocking model in mobile surveillance systems. In: Autonomous and Intelligent Systems (AIS), DOI 10.1109/AIS.2010.5547036

Mirzaei FM, Mourikis AI, Roumeliotis SI (2007) On the performance of multi-robot target tracking. In: IEEE International Conference on Robotics and Automation (ICRA)

Moors M, Rohling T, Schulz D (2005) A probabilistic approach to coordinated multi-robot indoor surveillance. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)

Moseley MB, Grocholsky BP, Cheung C, Singh S (2009) Integrated long-range UAV/UGV collaborative target tracking. SPIE Defense, Security, and Sensing

Mosteo AR (2010) Multi-Robot Task Allocation for Service Robotics : from Unlimited to Limited Communication Range. PhD thesis, Universidad de Zaragoza

Mottaghi R, Vaughan R (2007) An integrated particle filter and potential field method applied to cooperative multi-robot target tracking. Autonomous Robots 23(1):19–35

Murray RM (2007) Recent Research in Cooperative Control of Multivehicle Systems. Journal of Dynamic Systems, Measurement, and Control 129:571, DOI 10.1115/1.2766721

Murrieta R, Sarmiento A, Bhattacharya S, Hutchinson S (2004) Maintaining visibility of a moving target at a fixed distance: the case of observer bounded speed. In: IEEE International Conference on Robotics and Automation (ICRA), pp 479–484 Vol.1, DOI 10.1109/ROBOT.2004.1307195

Murrieta-Cid R, Gonzales-Banos HH, Tovar B (2002) A reactive motion planner to maintain visibility of unpredictable targets. In: IEEE International Conference on Robotics and Automation (ICRA)

Murrieta-Cid R, Muppirala T, Sarmiento A, Bhattacharya S, Hutchinson S (2007) Surveillance Strategies for a Pursuer with Finite Sensor Range. International Journal of Robotics Research (IJRR) 26(3):233–253, DOI 10.1177/0278364907077083

Murrieta-Cid R, Monroy R, Hutchinson S, Jea (2008) A complexity result for the pursuit-evasion game of maintaining visibility of a moving evader. In: IEEE International Conference on Robotics and Automation (ICRA)

Murrieta-Cid R, Ruiz U, Marroquin JL, Laumond JP, Hutchinson S (2011) Tracking an omnidirectional evader with a differential drive robot. Autonomous Robots 31(4):345–366, DOI 10.1007/s10514-011-9246-z

Nahin PJ (2012) Chases and Escapes: The Mathematics of Pursuit and Evasion (New in Paper). Princeton University Press

Nettleton E (2003) Decentralised architectures for tracking and navigation with multiple flight vehicles. PhD thesis, University of Sydney

Noori N, Isler V (2014) Lion and man with visibility in monotone polygons. International Journal of Robotics Research (IJRR) 33(1):155–181, DOI 10.1177/0278364913498291

Ohsumi A (1991) Optimal search for a Markovian target. Naval Research Logistics 38(4):531–554, DOI 10.1002/1520-

6750(199108)38:4¡531::AID-NAV3220380407¿3.0.CO;2-L

O'rourke J (1987) Art gallery theorems and algorithms. Princeton University Press

Oskam T, Sumner RW, Thuerey N, Gross M (2009) Visibility transition planning for dynamic camera control. In: SIGGRAPH/Eurographics Symposium on Computer Animation (SCA), ACM Press, New York, New York, USA, vol 1, DOI 10.1145/1599470.1599478

Papanikolopoulos NP, Khosla PK, Kanade T (1993) Visual tracking of a moving target by a camera mounted on a robot: A combination of control and vision. IEEE Transactions on Robotics and Automation 9(1):14–35

Parker LE (2002) Distributed Algorithms for Multi-Robot Observation of Multiple Moving Targets. Autonomous Robots pp 231–255

Parker LE, Emmons BA (1997) Cooperative Multi-Robot Observation. In: IEEE International Conference on Robotics and Automation (ICRA), pp 2082–2089

Parsons T (1978) Pursuit-evasion in a graph. Theory and applications of graphs

Pasqualetti F, Franchi A, Bullo F (2010) On optimal cooperative patrolling. In: IEEE Conference on Decision and Control (CDC), pp 7153–7158

Pimenta LC, Schwager M, Lindsey Q, Kumar V, Rus D, Mesquita RC, Pereira GA (2010) Simultaneous coverage and tracking (SCAT) of moving targets with robot networks. In: Algorithmic Foundation of Robotics VIII, Springer Tracts in Advanced Robotics, Springer Berlin / Heidelberg, pp 85–99

Pirjanian P, Mataric M (2000) Multi-Robot Target Acquisition using Multiple Objective Behavior Coordination. In: IEEE International Conference on Robotics and Automation (ICRA), pp 2696–2702

Pita J, Jain M, Ordónez F, Portway C (2009) Using game theory for Los Angeles airport security. AI Magazine pp 43–57

Portugal D, Rocha R (2010) MSP algorithm: multi-robot patrolling based on territory allocation using balanced graph partitioning. In: Symposium on Applied Computing, pp 1271–1276

Portugal D, Rocha R (2011) A survey on multi-robot patrolling algorithms. Technological Innovation for Sustainability pp 139–146

Portugal D, Rocha RP (2013) Distributed multi-robot patrol: A scalable and fault-tolerant framework. Robotics and Autonomous Systems DOI 10.1016/j.robot.2013.06.011

Pugh J, Martinoli A (2007) Inspiring and Modeling Multi-Robot Search with Particle Swarm Optimization. In: IEEE Swarm Intelligence Symposium (SIS), pp 332–339, DOI 10.1109/SIS.2007.367956

Qian K, Ma X, Dai X (2008) Simultaneous robot localization and person tracking using Rao-Blackwellised particle filters with multi-modal sensors. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp 22–26

Quigley M, Conley K, Gerkey B, Faust J, Foote T, Leibs J, Wheeler R, Ng AY (2009) ROS: an open-source Robot Operating System. ICRA Workshop on open source software

3(3.2)

R4SIM (2015) Workshop on Realistic, Rapid and Repeatable Robot Simulation, Robotics, Science and Systems conference, Roma (Italy)

Raboin E, Nau D, Kuter U, Gupta SK, Svec P (2010) Strategy generation in multi-agent imperfect-information pursuit games. In: International Conference on Autonomous Agents and Multiagent Systems (AAMAS)

Raboin E, Kuter U, Nau DS (2012) Generating strategies for multi-agent pursuit-evasion games in partially observable euclidean space. In: International Conference on Autonomous Agents and Multiagent Systems (AAMAS), pp 1201–1202

Raboin E, Svec P, Nau D, Gupta S (2013) Model-predictive target defense by team of unmanned surface vehicles operating in uncertain environments. In: IEEE International Conference on Robotics and Automation (ICRA), pp 3517–3522, DOI 10.1109/ICRA.2013.6631069

Renzaglia A (2012) Adaptive stochastic optimization for cooperative coverage with a swarm of Micro Aerial Vehicles. PhD thesis, Université de Grenoble

Riehl JR, Collins GE, Hespanha JP (2007) Cooperative graph-based model predictive search. In: IEEE Conference on Decision and Control (CDC), pp 2998–3004

Robin C, Lacroix S (2012) Failure anticipation in pursuit-evasion. In: Proceedings of Robotics: Science and Systems (RSS)

Roy N, Gordon G, Thrun S (2005) Finding approximate POMDP solutions through belief compression. Journal of Artificial Intelligence Research (JAIR)

Ruan S, Meirina C, Yu F, Pattipati K, Popp R (2005) Patrolling in a stochastic environment. Tech. rep., DTIC Document

Rybski PE, Stoeter SA, Erickson MD, Gini M, Hougen DF, Papanikolopoulos N (2000) A team of robotic agents for surveillance. In: Proceedings of the 4th International Conference on Autonomous Agents, ACM, pp 9–16

Sak T, Wainer J, Goldenstein SK (2008) Probabilistic multiagent patrolling. Advances in Artificial Intelligence pp 124–133

Santos J, Lima P (2010) Multi-robot cooperative object localization. In: RoboCup 2009: Robot Soccer World Cup XIII, pp 332–343

Sarmiento A, Murrieta-Cid R, Hutchinson S (2009) An Efficient Motion Strategy to Compute Expected-Time Locally Optimal Continuous Search Paths in Known Environments. Advanced Robotics 23(12):1533–1560, DOI 10.1163/016918609X12496339799170

Shi K, Cao Z, Zhang W, Zhou C (2010) The Targets Pursuit for Multi-robot System with Hybrid Wireless Sensor Networks. In: IEEE International Conference on Robotics and Automation (ICRA), pp 538–547

Soltero DE, Schwager M, Rus D (2013) Decentralized path planning for coverage tasks using gradient descent adaptive control. International Journal of Robotics Research (IJRR) DOI 10.1177/0278364913497241

Stone LD (2007) Theory of Optimal Search (2nd Edition). Academic Press New York, DOI 10.2307/2286890

Strom J, Morton R, Reilly K, Olson E (2010) Online proba-

bilistic pursuit of adversarial evaders. In: ICRA Workshop: Search and Pursuit/Evasion in the Physical World

Svec P, Thakur A, Raboin E, Shah B, Gupta S (2014) Target Following with Motion Prediction for Unmanned Surface Vehicle Operating in Cluttered Environments. Autonomous Robots 36(4):383–405, DOI 10.1007/s10514-013-9370-z, URL http://dx.doi.org/10.1007/s10514-013-9370-z

Tang Z, Ozguner U (2005) Motion planning for multi-target surveillance with mobile sensor agents. IEEE Transactions on Robotics 21(5):898–908

Tanner HG (2007) Switched UAV-UGV cooperation scheme for target detection. In: IEEE International Conference on Robotics and Automation (ICRA)

Theodorakopoulos P (2009) On autonomous target tracking for UAVs. PhD thesis, University of Toulouse

Tsokas Na, Kyriakopoulos KJ (2011) Multi-robot multiple hypothesis tracking for pedestrian tracking. Autonomous Robots 32(1):63–79, DOI 10.1007/s10514-011-9259-7

Urrutia J (2000) Art gallery and illumination problems. Handbook of computational geometry

Vidal R, Shakernia O, Kim HJ, Shim DH, Sastry S (2002) Probabilistic PursuitEvasion Games: Theory, Implementation, and Experimental Evaluation. IEEE Transactions on Robotics and Automation 18(5):662–669

Vieira M, Govindan R, Sukhatme G (2009) Scalable and practical pursuit-evasion with networked robots. Intelligent Service Robotics

Vo C, Lien Jm (2010) Visibility-Based Strategies for Tracking and Searching Unpredictable Coherent Targets Among Known Obstacles. In: ICRA Workshop: Search and Pursuit/Evasion in the Physical World

Wang Z, Gu D, Meng T, Zhao Y (2010) Consensus Target Tracking in Multi-robot Systems. Intelligent Robotics and Applications pp 724–735

Wong Em, Bourgault F, Furukawa T (2005) Multi-vehicle Bayesian Search for Multiple Lost Targets. In: IEEE International Conference on Robotics and Automation (ICRA), pp 3180–3185

Xu Z, Douillard B, Morton P, Vlaskine V, Wohlleber C, Calleija M, Underwood J, Sukkarieh S (2012) Towards Collaborative Multi-MAV-UGV Teams for Target Tracking. In: Proceedings of Robotics: Science and Systems (RSS), Workshop on Integration of perception with control and navigation for resource-limited, highly dynamic, autonomous systems

Xu Z, Fitch R, Sukkarieh S (2013) Decentralised Coordination of Mobile Robots for Target Tracking with Learnt Utility Models. In: IEEE International Conference on Robotics and Automation (ICRA), pp 2006–2012

Yau J, Chung TH (2012) Search-theoretic and Ocean Models for Localizing Drifting Objects. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)

Yu H, Beard R (2011) Probabilistic path planning for cooperative target tracking using aerial and ground vehicles. In: American Control Conference (ACC), pp 4673–4678

Zhou K, Roumeliotis SI (2011) Multirobot active target tracking with combinations of relative observations. IEEE Transactions on Robotics 27(4):678–695, DOI 10.1109/TRO.2011.2114734

Zhou PC, Hong BR, Wang YH, Zhou T (2004) Multi-agent coopoerative pursuit based on extended Contract Net Protocol. In: Machine Learning and Cybernetics