Search and Pursuit-Evasion in Mobile Robotics

A survey

Timothy H. Chung · Geoffrey A. Hollinger · Volkan Isler

Abstract This paper surveys recent results in pursuit-evasion and autonomous search relevant to applications in mobile robotics. We provide a taxonomy of search problems that highlights the differences resulting from varying assumptions on the searchers, targets, and the environment. We then list a number of fundamental results in the areas of pursuit-evasion and probabilistic search, and we discuss field implementations on mobile robotic systems. In addition, we highlight current open problems in the area and explore avenues for future work.

The authors have been listed alphabetically based on equal contribution to the article.

Timothy H. Chung
Department of Systems Engineering
Graduate School of Engineering and Applied Sciences
Naval Postgraduate School
Monterey, CA USA
Tel.: +1-831-656-7858
E-mail: thchung@nps.edu

Geoffrey A. Hollinger
Computer Science Department
Viterbi School of Engineering
University of Southern California
Los Angeles, CA USA
Tel.: +1-904-993-1584
E-mail: gahollin@usc.edu

Volkan Isler
Department of Computer Science and Engineering
College of Science and Engineering
University of Minnesota
Minneapolis, MN USA
Tel.: +1-612-625-1067
E-mail: isler@cs.umn.edu

1 Introduction

Joint research between divergent disciplines has led to significant advances in autonomous search and pursuit-evasion with mobile robots. While robotics applications have often served as catalysts for vibrant research at the intersection of traditional disciplines, only recently have researchers undertaken the study of robotic systems for search missions and pursuit-evasion contexts. This article surveys recent advances in this area, which leverage both theoretical foundations and practical implementations to forge new and innovative results.

Search and pursuit-evasion problems (also known as "one-sided search" and "adversarial search," respectively) have traditionally been addressed using two contrasting approaches. One perspective has been to design strategies that maximize searcher performance against a worst-case adversary. In such settings, the evader is often characterized by infinite speed, complete awareness of searcher location and intent, and full knowledge of the search environment. Such methods offer guarantees on the success of the search, defined, for example, by capture of the target in finite time. However, the powerful adversary model may yield solutions that are too conservative in practical applications. In contrast, parallel research has emphasized probabilistic formulations addressing average-case behaviors. Measures of interest can include expected time until detection or expected number of glimpses. The assumption about knowledge about the evader behavior allows incorporating probabilistic uncertainty in target locations, their behaviors, and/or sensor observations.

Many variations on the theme of search and pursuit-evasion problems exist, due to the diverse contexts in which they are studied. Figure 1 outlines a partial taxonomy of the parameter space for search and pursuit-

evasion models. Two prior surveys in pursuit-evasion have focused on adversarial search when the environment is represented as a graph (Alspach, 2004; Fomin and Thilikos, 2008). These surveys do not include probabilistic search or field implementations on mobile robots. In the current paper, we focus on the connection between fundamental work in search and pursuit-evasion and applications to related problems in mobile robotics. Rather than presenting a broad survey of results on a wide range of pursuit-evasion games (which would require volumes), we limit our focus to a small number of games which, in our opinion, are directly related to robotics. We present results on the variants of these games based on, e.g., the complexity of the environment and the sensing powers of the players, both of which are important in modeling realistic scenarios. We also omit the large body of work on related differential games, such as the homicidal chauffeur game, and refer the reader to the book on this topic (Başar and Olsder, 1999).

Similarly, though works in classical search theory (Benkoski et al., 1991) construct stochastic optimization models for operational settings, these approaches often face shortcomings when addressing the computational, sensing, and mobility questions arising in physical mobile robot systems. We specifically examine the foundational elements of search theory that have commonalities with relevant robotic missions, such as the desire to maximize detection probabilities spatially or minimize time until detection in temporal contexts. This survey aims to bridge these related communities through their respective but relevant formulations and theoretical approaches.

This article surveys recent works in search and pursuit-evasion research with applications to mobile robotic systems. We have limited the scope of the paper to two types of problems: (1) adversarial pursuit-evasion games on graphs and in polygonal environments and (2) probabilistic search where the motions of the searchers and targets are independent. To see the difference between the two formulations, imagine a pursuer and evader in a room with a round obstacle in the middle. To make the game precise, let us assume that the players can see each other at all times and have equal maximum speeds. Suppose the pursuer picks his initial location, followed by the evader. In this scenario, it is easy to see that there exists an evader strategy which avoids capture indefinitely. Therefore, we say that a single pursuer cannot capture an evader in this environment. In contrast, suppose the evader is moving in the same environment in a way that is independent of the pursuer's motion. For example, it may be performing a random walk. In this case, the evader can be captured. In fact,

capture of the randomly walking evader can occur even with a stationary pursuer, as the evader will eventually hit the pursuer! The main question here would be the design of an optimal strategy to capture the randomly walking evader as quickly as possible. This is the focus of probabilistic search.

In Section 2 we present results on a number of fundamental pursuit-evasion games that take place either on graphs or in polygonal environments. The problems in this section are games because the players have conflicting objectives: the pursuers try to "capture" the evaders, and the evaders actively avoid capture. In Section 3, we focus on probabilistic search problems in which the target motion is independent of the pursuer strategy. In Section 4, we discuss transitional research that bridges the gap between theory and relevant applications on robotic systems. We then discuss problems that arise specifically from robotics applications, and we highlight challenges faced in implementations and field studies. Finally, in Section 5, we conclude the survey with a discussion of open problems and avenues for future research.

2 Pursuit-Evasion Games

In a pursuit-evasion game, one or more pursuers try to capture one or more evaders who, in turn, try to avoid capture. In robotics, pursuit-evasion games are used for studying motion planning problems that arise in adversarial settings, such as catching burglars, playing hide-and-seek, and so on. In addition, pursuit-evasion games are used to obtain results on the worst-case performance of robotic systems. For example, imagine a search-and-rescue setting in which the robots try to find a lost person. In this case, by treating the lost person as an adversarial entity trying to avoid being found, one can obtain worst-case bounds on the number of robots necessary for rescue in a given environment. This is because a pursuit strategy, if it exists, will guarantee that the person would be found no matter how s/he moves. On the other hand, in some settings a pursuit-evasion formulation can be too conservative, for example, in terms of the number of pursuers. In such cases, a probabilistic model (Section 3) can be more appropriate.

Beyond robotics, pursuit-evasion games find applications in numerous other settings (network security, modeling animal behavior, just to name a few). As a result, researchers have studied many different versions of pursuit-evasion games based on:

- Environment where the game is played: Examples include plane, grid, graph, polygons, etc.
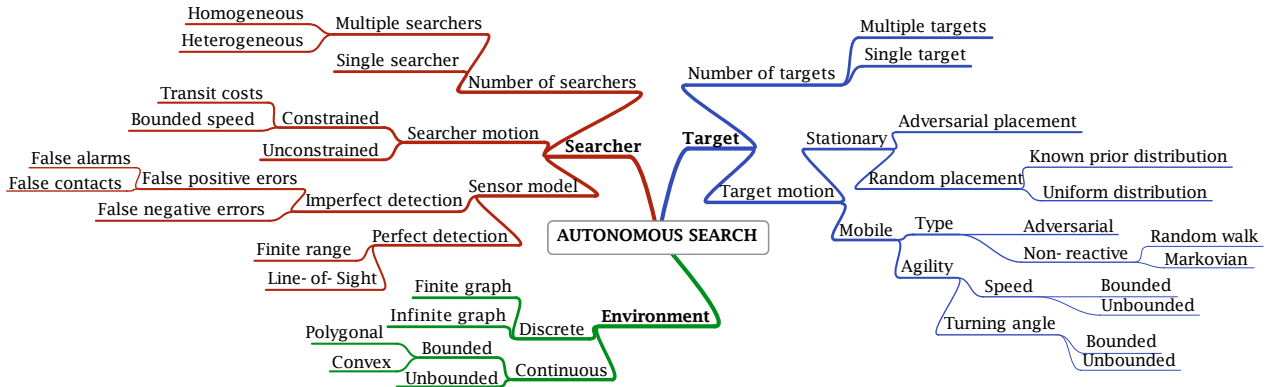
Fig. 1 Illustration of the parameter space for autonomous search models.

- Information available to the the players: Do they know each other's positions all the time?
- Controllability of the players motion: Is there a bound on their speed? Can they turn with arbitrary angles?
- Meaning of capture: In some games, the pursuer captures the evader if the distance between them is less than a threshold. In other games, the pursuers must see or surround the evader in order to capture it.

In robotics literature, there are two primary approaches for solving pursuit-evasion games: differential and combinatorial. The former approach is based on techniques developed for solving non-cooperative differential games (cf. Başar and Olsder (1999)). In these games, differential equations governing the motion of the players are brought together using Hamilton-Jacobi-Isaacs (HJI) differential equations. The solutions to HJI equations yield players' strategies as control inputs for achieving capture or evasion objectives. The advantage of this approach is that one can model physical constraints such as bounds on turning velocity or acceleration by expressing them as differential constraints. On the down side, since the resulting equations are rather complex (especially in complex environments), their solutions are often numerical, only locally valid, and/or heuristic-based rather than globally optimal.

Games that take place in complex environments (the main focus of this section) are usually solved using combinatorial techniques. A common approach is to represent the environment geometrically (e.g., with a polygon) and solve the game directly using this representation. Alternatively, a graph can be used to represent the environment topologically. For example, one can use a graph whose vertices correspond to rooms and corridors in a building. The edges will then correspond to doors and represent connections between rooms and corridors. In general, one can start with a geometric representation and study the pursuit-evasion game which takes place on a graph extracted from the geometric features. Natural candidates for such graphs are the dual of the triangulation of a polygon or its medial axis.

In the remainder of this section, we survey pursuit-evasion games that take place in these two types of settings, and list open problems. We start with games that take place on graphs.

2.1 Pursuit-Evasion on Graphs

In this section, we review work on two fundamental pursuit-evasion games that take place on graphs. In the cops-and-robbers game (Section 2.1.1), the players can move one edge at a time. In contrast, the single evader in Parson's game can be arbitrarily faster than the pursuers (Section 2.1.2).

2.1.1 The Cops and Robbers Game

One of the basic games that takes place on graphs is the cops and robbers game. In this game, the cops (pursuers) try to capture a robber (evader) by moving along the vertices of a graph. The players move in turns along the edges. The cops win the game if they can move onto the robbers vertex. This game was introduced by Nowakowski and Winkler (1983), and by Aigner and Fromme (1984).

There are two immediate questions. (1) Given a graph, what is its *cop number*: the minimum number of cops necessary to capture the robber regardless of the initial locations of the players? (2) What is the class of graphs whose cop number is a given number? Let us first consider a simple algorithmic solution, based on the dynamic programming principle, for this game which will be useful in highlighting its interesting aspects.

Let $G = (V, E)$ be the graph where the game takes place. For each vertex $v \in V$, $N(v)$ denotes its neighborhood: $N(v) = \{u : (u, v) \in E, u, v \in V\}$. For now, we assume that there is a single cop. We will represent the state of the game with a pair (c, r) corresponding to the cop's and the robber's locations respectively. Consider the following algorithm which iteratively marks the states of the game. Initially all states are unmarked.

Marking Algorithm for Cops-and-Robbers

- For all $u \in V$, mark the state (u, u)
- repeat
    - for all unmarked states (c, r),
      if $\forall r' \in N(r), \exists c' \in N(c)$ such that (c', r') is marked, then mark (c, r).
- until no further marking is possible

It is a simple exercise to prove the statement "if all states are marked, the cop can win the game from any starting state" by induction on the marking order. For the other direction ("if there is an unmarked state, the robber wins game"), suppose that there is an unmarked state (c, r) and the players start the game at vertices c (for the cop) and r (for the robber). It must be that there is a vertex $r' \in N(r)$ such that if the robber moves to r', no matter which vertex c' the cop moves to, the resulting state (c', r') must be unmarked; otherwise (c, r) would be marked. Therefore, the robber can force the game to stay in an unmarked state. Since all the capture states are initially marked, this means that the cop can never capture the robber.

Now that we have a simple algorithm to solve the single cop game, let us consider various extensions.

Number of cops: A naïve implementation of the algorithm above would run in $O(n^4)$ steps, where $n = |V|$. Now, suppose there are k cops in the game. By extending the state representation from pairs of the form (c, r) to $(k+1)$-tuples of the form $(c_1, \ldots, c_k, r)$, the algorithm above can be extended for the multi-cop case. However, the state space is now of size $O(n^{k+1})$, which causes the running time to increase to $O(n^{2(k+1)})$. This exponential increase in the running time clearly makes the algorithm impractical for a large number of cops.

Of course, it is possible that there are more efficient algorithms for solving the cops and robber game. Indeed, for the single cop version, more efficient algorithms were presented both by Nowakowski and Winkler (1983), and Aigner and Fromme (1984). However, the problem of determining whether k cops with given initial locations can capture a robber on a given undirected graph is EXPTIME-complete (Goldstein and Reingold, 1995). Therefore, it is not likely that significantly more efficient algorithms exist for the general case.

One approach for dealing with this complexity issue is to limit the class of graphs on which the game is played. Structurally, it is easy to see that the cop number of trees and cliques is one. Aigner and Fromme (1984) show that the cop number of planar graphs is at most three. Another approach is to approximate the minimum number of cops. This interesting aspect of the problem has not received significant attention which gives us our first open problem: *Are there efficient algorithms for approximating the cop-number of a given graph?*

Order of Play: Imagine a game between a single cop and the robber on a complete graph K_n. If the players move in turns, the cop can capture the robber in a single move. If they move simultaneously, the game does not admit a deterministic solution in the sense that if the cop fixes a deterministic strategy based on the location of the robber, the robber can avoid capture indefinitely. Similarly, if the robber fixes his strategy, he will be captured after a single move. Of course, there is a mixed equilibrium in which the players pick their next location uniformly at random which yields capture in n steps.

It seems that randomization can be used to solve simultaneous-move games in general. As an example, consider the game on a dismantlable graph on which a single cop wins under the turn-based model. It can be shown that the duration of the game is bounded by n, the number of vertices (Isler and Karnad, 2008). Hence, the cop can guess the next n moves of the robber and capture him with a small but non-zero probability. One would expect that there are more efficient strategies: In the turn-based model, when players move optimally, the last configuration on a dismantlable graph before capture is (c, r) with $N(r) \subset N(c)$. At this stage, the cop can simply guess the move of the robber and capture him. Turning this intuition into an efficient randomized strategy is not straightforward since it is not obvious how the cop can force this end condition under the simultaneous move model.

One helpful observation is that whenever the cop can capture the robber in the turn-based model, he has a strategy to reduce the distance between the players to one in the simultaneous move model. In the latter case, the cop simply waits for a single step and executes the turn-based strategy.

We summarize the discussion with the following open problems: *Characterize the role of the order of play in the cops-and-robbers game. Is it true that a strategy for the turn-based model can be turned into an efficient randomized strategy under the simultaneous-move model?*

The order of play has interesting implications when the underlying domain is continuous. This issue is further discussed in Section 2.2.

Information available to the players: In the basic cops-and-robbers game, it is assumed that the play-

ers know each other's positions at all times. There are some settings in which this assumption is plausible. For example, Vieira et al. (2009) developed a system where a network of stationary sensors provide information about the location of the evader to the pursuers. In such settings, the basic cops-and-robbers formulation is applicable. However, in most robotics applications, the players must operate under sensing limitations.

As a start, imagine that the players can not observe each other unless they are located on the same vertex. This game is known as the *hunter-and-rabbit game*. It has been studied by Adler et al. (2003) [1]. Let us revisit the game played on the complete graph K_n. Consider any deterministic pursuer strategy which can be specified, for example, by an ordered list of vertices to be visited. As soon as the pursuer fixes this strategy, the evader can avoid the pursuer by picking a different order. In this case, the evader does not need to "see" the pursuer. By simulating the pursuer's strategy, the evader will know the pursuer's position at all times. A symmetric argument can be made for deterministic evader strategies. Using the standard game theory terminology, we have just argued that this game does not have a solution for pure (deterministic) strategies. Observe that the pursuer can win the game (with high probability) simply by picking the next vertex uniformly at random. Since the evader can not infer the pursuer's next move, the pursuer's probability of success at each move is $\frac{1}{n}$. Therefore, the pursuer is expected to capture the evader in n trials. The probability of not capturing the evader after $n \ln n$ trials is upper bounded by $\frac{1}{n}$. It is worth emphasizing that a randomized strategy where the player randomizes his own actions is different from probabilistic games (presented in the next section) which are solved under the assumption that the other player is moving according to a probabilistic model (e.g., a randomly walking evader).

In most cases, the solution of a limited information game is randomized. Note that this argument is environment-specific and a randomized strategy is not always necessary. For example, the pursuer can sweep a path (graph) and capture the evader using a deterministic strategy. Nevertheless, Adler et al. (2003) showed that in the no-visibility version, a single pursuer can catch the evader in $O(n \log n)$ expected time on *any* graph. It was also shown that this analysis is tight: there are graphs and matching evader strategies which guarantee that no pursuer strategy can capture the evader in less than $\Omega(n \log n)$ steps in expectation.

So far, we considered the two extremes regarding the visibility of the players: global (i.e., full) visibility

and no visibility. When the players have limited (or local) visibility, representing the state of the game in a compact fashion becomes harder. To see this, let us revisit the marking algorithm given above. Suppose a single cop tries to capture the robber. As the game progresses, the robber will keep track of a set C of *all possible locations* of the cop. Of course, when the robber sees the cop, C will contain only the cop's location. As the players move, set C can contain multiple vertices. Now, imagine that the robber is at location r, the cop is not visible, and for every position r' the robber can move to, there is a vertex $c \in C$ from which the cop can land on r'. If this happens, the cop can obtain non-zero capture probability by randomizing among strategies that lead to vertices in C. Hence, a winning robber strategy, if it exists, must prevent the game from entering configuration (r, C). Observe that the state of the game now includes sets of vertices (C) which may result in an explosion in the number of states. This makes designing dynamic programming based algorithms for solving pursuit-evasion games with limited visibility difficult.

Isler et al. (2006) studied the case where the evader has local visibility. They study a variant where the players move simultaneously, and introduce the notion of i-visibility where a player with i-visibility can see another player only if the distance between them is at most i. It was shown that when the evader has 1-visibility (i.e., can see only the neighbors of its current location), two cops with 1-visibility can capture the evader with high probability on any graph. The expected capture time with two pursuers is polynomial in the number of vertices. A characterization of cop-win graphs where a single pursuer suffices to capture the evader was also presented. It was also shown that when the evader has 2-visibility, the number of cops required becomes unbounded: there are graphs which require $\tilde{\Omega}(\sqrt{n})$ cops to capture an evader with 2-visibility.
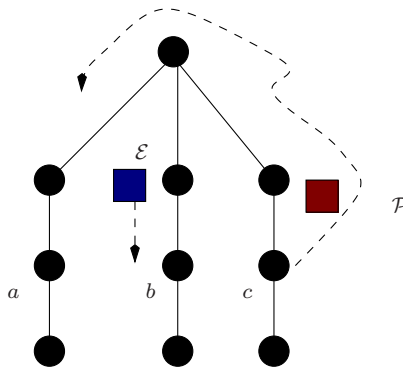
Fig. 2 Both players have 1-visibility. On this graph, the pursuer $\mathcal{P}$ can not capture the evader $\mathcal{E}$ using a deterministic strategy. However, a randomized capture strategy exists. (From Isler et al. (2006)).

---

[1] To avoid confusion, hereafter we will use the term pursuer for the cop/hunter and evader for the robber/rabbit.

Similar to the no-visibility case, the winning pursuit strategies are often randomized. A simple example is illustrated in Figure 2. Suppose, on this graph, the pursuer has a deterministic strategy of visiting the labeled vertices in the order a, b, c. Then, we can design a evader strategy that waits until the pursuer arrives at b and escapes to a. Afterwards, while the pursuer is visiting c, the evader escapes to b and it is easy to see that by repeating similar moves, the evader can always avoid the pursuer. However, on this graph there is a simple randomized strategy for the pursuer: pick one of the leaves at random and visit that leaf.

More recently, Isler and Karnad (2008), studied the effect of reducing the pursuer's (i.e., the cop's) visibility against an evader with global visibility. Let $\mathcal{G}$ be the class of graphs where a single pursuer with global visibility can capture the evader. They showed that a pursuer with limited visibility can capture the evader with high probability on any graph in $\mathcal{G}$. However, there exist graphs where the capture time is exponential in the number of vertices. This was proven constructively by presenting evader strategies which guarantee that the expected capture time is lower-bounded by an exponential quantity.

Many open problems remain regarding identifying the role of information available to the players on the outcome of the game. We list the most general versions as open problems: *Suppose the cops have i-visibility and the robber has j-visibility; what is the class of graphs on which k cops suffice? What is the minimum number of cops necessary to capture the robber on a given graph?*

2.1.2 Parson's game

When the graph on which the game takes place is an abstraction of a geometric environment, the temporal aspect of the game is sometimes lost. For example, in the cops-and-robbers game, players move along one edge at a time. But if the edges correspond to corridors of varying lengths, the turn-based model fails to capture this variability. One approach, in line with the worst-case spirit of pursuit-evasion games, is to make the evader very powerful and to treat it as a adversarial target which can be arbitrarily faster than the pursuers.

In this "infinite speed model," the graph can be considered as a system of tunnels represented by the edges of the graph in which an evader is hiding. To catch such a powerful evader, the pursuers must surround it. Alternatively, one can view the evader as a harmful gas contaminating the graph which must be cleared. Parsons (1976) defined the *search number* (or $sn(G)$) of a graph to be the minimum number of pursuers necessary for capture. Determining the search number of a

graph was later found to be NP-hard (Megiddo et al., 1988), and to be NP-complete due to the monotonicity of optimal edge search schedules (Bienstock and Seymour, 1991; LaPaugh, 1993). In this work, the evader can only hide in the edges of the graph (referred to as *edge search*).

Various versions of the infinite speed model play a key role in solving visibility-based search problems in robotics. These games will be reviewed in Section 4.

2.2 Pursuit-Evasion in Geometric Settings

In this section, we review the lion-and-man game which is perhaps the most natural game to capture robotics applications. After an overview of basic results, we present results on variants of this game.

The *lion-and-man game* is a geometric version of the cops-and-robbers game. In the original version, the game takes place inside a circular arena of radius r. The players have the same maximum speed, which we will assume to be one. The objective of the lion (pursuer) is to capture the man by moving onto the his current location. Suppose the players move in turns. It is not too difficult to see that the lion can win the game with the following strategy: starting from the center of the circle, stay on the radius that passes through the man's current location and move as close as possible to the man (Figure 3-left). It can be shown that the capture time of this strategy is $O(r^2)$ using a similar analysis to the one given by Sgall (2001). It is interesting to note that when the players move in continuous time, this strategy does not reduce the distance between the players to zero in finite time. Littlewood (1953) provided a proof of how a similar argument can be made against any lion strategy. Therefore, if the game is played in continuous time, the man escapes. Alonso et al. (1992) showed that there exists a strategy with which the lion can get within a distance c of the man in time $O(r \log \frac{r}{c})$. They also show that this results is almost tight by presenting an evasion strategy for the man which delays capture for at least $\Omega\left(r\sqrt{\frac{log(r/c)}{\log \log(r/c)}}\right)$ steps.

Sgall (2001) studied a version of the lion and man game which takes place in the first quadrant (Figure 3-middle). Let M be the initial location of the man, and L be the initial location of the lion. If one of the coordinates of the man is greater than or equal to the corresponding coordinate of the lion, the man wins the game. In the remaining case, Sgall shows that the lion can win the game as follows: Let R be the line passing through L and M. The lion finds the smallest circle which touches both of the axes and the center C of the circle is on R. Afterwards, when the man moves to M',
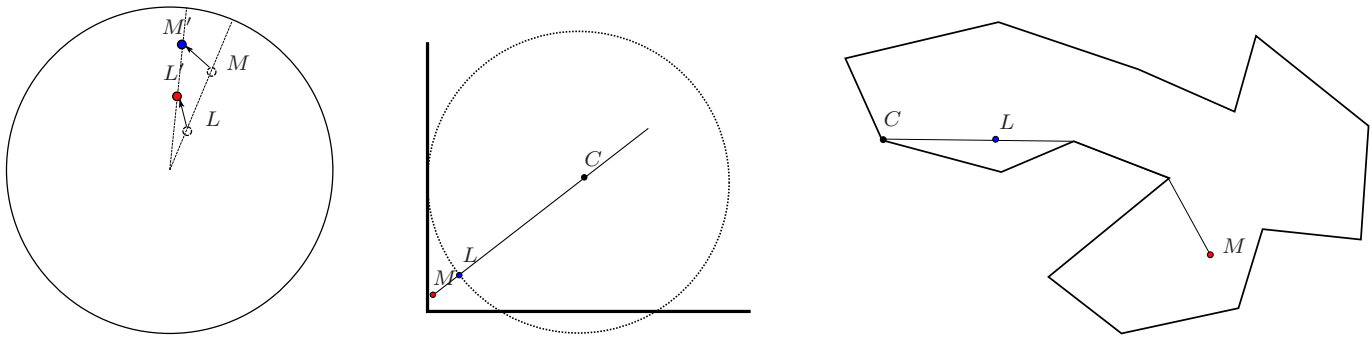
Fig. 3 Lion and man in various environments. In all three cases, the lion guards a region given by a fixed center and the lion's position and makes progress by making the guarded region larger. **Left:** solution to the original lion and man game. **Middle:** lion and man in the first quadrant. **Right:** lion and man in polygons.

the lion moves to the point closest to the man on the line $M'C$ that it can reach. The number of the moves required for this strategy is quadratic in the lion's initial distance to the origin, but it is also a function of the slope of the line R. Note that the players move in turns as opposed to the model used in the original version where they move simultaneously in continuous time.

Isler et al. (2005) studied the lion-and-man game in polygonal environments (Figure 3-right) and showed that a single lion can capture the man in any simply-connected polygon. Recently, Bhadauria and Isler (2011) showed that three lions can capture the man in any polygon (possibly with holes). This leaves the following problems open: *What is the class of polygons in which two lions can capture the man? What is the number of lions necessary to capture the man in a complex environment when the players are subject to sensing limitations?*

Variations: Perhaps the first strategy any one would think of for the lion and man game is the greedy strategy: move toward the man. One of the advantages of the greedy strategy is that it is easy to implement. Further, in the circular case, it seems that the lion can eventually capture the man using the greedy strategy. This is because, to maintain the separation between the players, the man has to move away from the lion along the line connecting them. This can not go on forever: Eventually he will have to turn because of the boundary. At this point, the distance between the players decreases. However, the gain in distance can be arbitrarily small for example when both players are located on the boundary of a large circle.

Recently, researchers studied two variations of the lion and man game: higher dimensions and sensing limitations. On the open plane, it has been known that three lions can capture the man if and only if the man's initial location is contained in their convex hull (Jankovic, 1978). Recently, Kopparty and Ravishankar (2005) generalized this result to arbitrary dimensions: $d + 1$ lions

suffice in R^d. Alexander et al. (2009) further generalized this result to convex environments of arbitrary dimension and shape. The same authors studied the game in bounded environments and showed that the lion can decrease the distance between the players by executing the greedy strategy if the environment has the so-called CAT(0) property (Alexander et al., 2006).

More recently, Bopardikar et al. (2007) studied sensing limitations in the lion-and-man game. In their model, the lion can observe the man's location only if the distance between the players is less than a given threshold. The man is *reactive* in the sense that he moves only when he sees the lion. They show that the lion strategy outlined in Figure 3 can be used to capture such a reactive evader. Karnad and Isler (2008) focused on a different type of sensing model in which the lion can measure only the man's bearing angle. In their model, the players move in turns and take measurements after each move. They studied the game in the first quadrant and showed that the lion can reduce the distance between the players to the step size. The lion and man game in a general three-dimensional environment remains open.

3 Probabilistic Search

Pursuit-evasion games, presented in the previous section, seek to maximize the worst-case performance on search or capture. In contrast, probabilistic search methods consider optimization of the expected value of a search objective, such as maximal probability of detection or minimal time to detection. The latter class of probabilistic approaches is the focus of the theory of search, which has a long-standing legacy in the field of Operations Research (OR). Bernard Koopman's seminal works in search theory outlined analytic principles for applied probability and optimization models for maritime warfare strategies (Koopman, 1956a,b, 1957)

This original formulation required analytical specification of the following components of a search-theoretic model: (i) an *a priori* probability distribution of the target's location within the search region, (ii) a measure of the density of search effort, and (iii) the detection probability afforded by application of a given density of search effort. For example, search in a plane (such as the open ocean) requires a two-dimensional target probability density (i.e., probability per unit area), denoted $p(x, y)$, with detection probability defined as

$$D(x, y, z) = \Pr[\text{target detected}|\text{target at } x, y$$
$$\text{and search effort } z].$$

The search effort can be quantified, often interchangeably, by the search time, e.g., hours spent inspecting a sector, or by the search cost, e.g., resources expended observing an area.

In his book, McCue (1990) provides a thorough and illustrative case study outlining an extensive operations analysis of the search operations conducted during World War II for submarines in the Bay of Biscay. In addition to mathematical models governing the likelihood of target presence, McCue quantitatively models the density of the search effort in the context of the searcher's sensor sweep width (visual and radar detectors) and flight speed. Such values define the sweep rate, which, in conjunction with the chosen flight search path and its length (or search duration), can determine the probability of detecting a submarine within the sensor's sensor swath. This text highlights the various modeling elements utilized to apply the probabilistic search techniques, and its relevance to robotic search and pursuit-evasion applications is clear.

This probabilistic model forms the foundation of the theory of search. Since the total search effort, such as search time or physical assets, is often a constrained resource, the question of optimal allocation of the search effort is of interest. Koopman further developed an exponential model for detection of targets based on random search, which adequately reflects realistic sensors and also offers analytic advantages. The exponential model offers a mathematical representation of improving detection probabilities, while reflecting diminishing returns with increasing search effort. Exploration of these combined probability and optimization models engendered significant research in both applied math and the OR communities. Such investigations include extensions to the probabilistic descriptions of the prior distributions of target locations throughout a spatial region of interest, as well as variations on the detection models which account for the possibility of overlooking the targets by the searcher's imperfect sensors. For these search problems, the objective is often either to minimize the expected time until a target is detected or to maximize the probability of its detection in the face of "missed detection" probabilities.

Dobbie (1968) provided a survey of the classical search theory literature, the first of several periods in search theory, according to Stone (1989b), labeled as *classical* (1942-1965), *mathematical* (1965-1975), *algorithmic* (1975-1985), and *dynamic* search (1985-present). The classical period describes the development of simple, yet practical, models for search and detection, enabling immediate application to search problems of interest. Emphasis on analytic rigor characterized the mathematical era, with focus placed on developing necessary and/or sufficient conditions on the optimality of search plans. As cheap computational resources evolved, so did the research in the algorithmic period of search theory, where numerical methods were constructed for addressing more complex search problems, such as those involving moving targets. Results from all of these preceding eras provide the fundamental methods for addressing dynamic search problems, where information acquired during the search process is readily incorporated and used to generate new optimal search plans. This last class of search problems continues to present interesting and open challenges, increasingly so as greater employment of autonomous systems have broadened the communities of interest.

Benkoski et al. (1991) offered an annotated bibliography, in which the authors provide an abridged taxonomy for the search theoretic problems to date, by first dividing the literature into one-sided search and search games and then investigating a number of subdivisions therein, illustrated by Figure 4. The former relates to contexts where the target is unaware of the search process and does not act to conceal its whereabouts nor actively evade an approaching searcher. The optimization of search is unopposed, which may apply in operations where the searcher's ability for stealth surpasses that of the target. On the other hand, search games, as described here, have relevance to the adversarial behaviors examined in the previous section. However, many works in this category also consider discrete time and space, including stationary placement of a target to adversely affect the searcher's efforts.

Much of these early works on the theory of optimal search relied on assumptions to facilitate the formal analysis, including exponential detection models and continuous allocations of search effort (Koopman, 1979). Stone consolidated these analytic formulations and developed conditions for uniformly optimal search plans, relying on the above assumptions for sensing with missed detections, continuous searcher trajectories, and
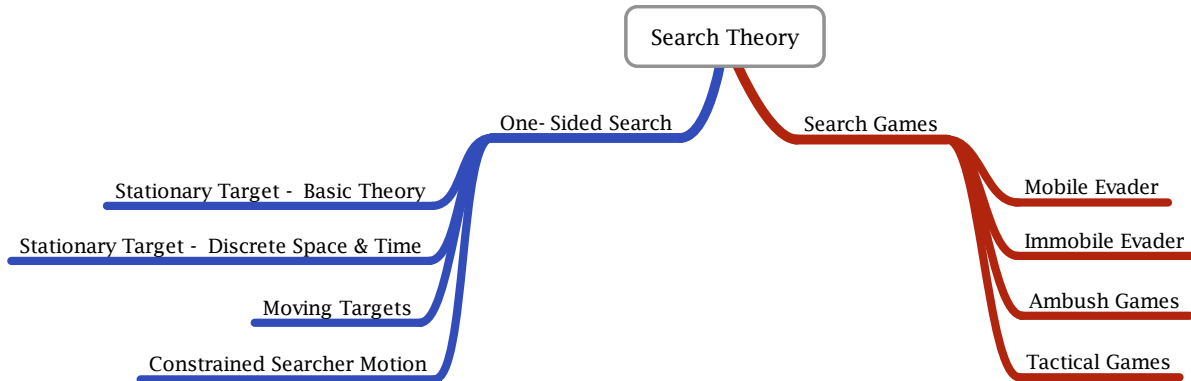
Fig. 4 Illustration of the partial taxonomy provided by Benkoski et al. (1991) for search theoretic problems and application areas previously considered in the operations research and applied mathematics communities.

divisible search effort, and solved using Lagrange multipliers (Stone, 1989a).

Other search theoretic works during these earlier eras involve characterization of lateral range curves for detection, e.g., for both sonar and visual sensors (see Figure 5), developing empirically-validated models for physical sensing phenomena (e.g., passive and active acoustic sensing) (Wagner, 1999), and development of bounding cases such as random search in the presence of overlook probabilities (Washburn, 2002) (see Figure 6). Though much of these sensing models remain valid for contemporary operational search problems (e.g., anti-submarine warfare), there has been rapid development of new sensor technologies, such as 3D LIDAR, depth cameras, synthetic aperture sensors, etc., and their new capabilities, including fused sensing modalities. These systems offer the following open problem: *Can unified analytic models be developed for modern and future sensing modalities that are applicable to probabilistic search theoretic formulations?*

The emphasis on continuous space and search effort historically offered a context for analytic tools, such as the calculus of variations, to solve for optimal search distributions. Examination of discrete search space, either via partitioning of a continuous area or by the inherent nature of the search problem (e.g., search for an object among n discrete boxes), also developed as an active area of research, even more so as computational capabilities have evolved. In these models, suppose the search for the target occurs over n locations or cells, such that there is some finite probability, p_i, that the target resides in the ith cell. Let the detection function $D(i, z)$ be the probability of detecting the target in cell i given that the target is truly in cell i and z search
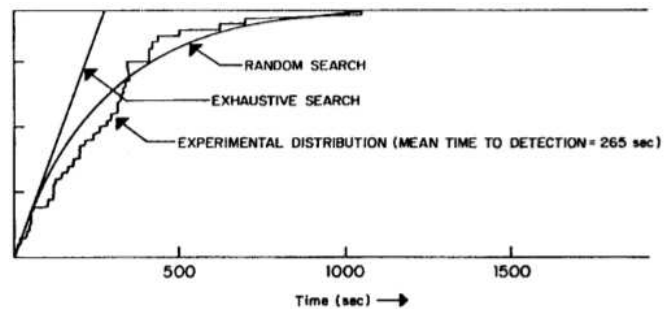
Fig. 6 Illustration (from Washburn (2002)) of the cumulative density function for the time until target detection for the random search model. Search strategies which seek to, for example, minimize the expected time until target detection, can be evaluated and benchmarked against the random search model, which serves as a lower bound on search performance.

effort has been applied. The search effort z can still be a continuous (e.g., time spent inspecting the cell) quantity or can now be represented in discrete units (e.g., number of glimpses or inspections of the cell). As before, the metric remains to either maximize the detection probability or to minimize the time until the target is found.

Some of the key results include the characterization of the computational complexity of the constrained path optimal search problem, where the searcher incurs a penalty (e.g., travel time or cost) for transit between locations in the search region. Wegener (1985) showed that, for positive penalties or "switching costs," even special cases of the optimal search problem are NP-hard. Trummel and Weisinger (1986) later independently generalized these complexity results by reduction from finding Hamiltonian paths on a graph, finding that maximization of the probability of detection is an NP-
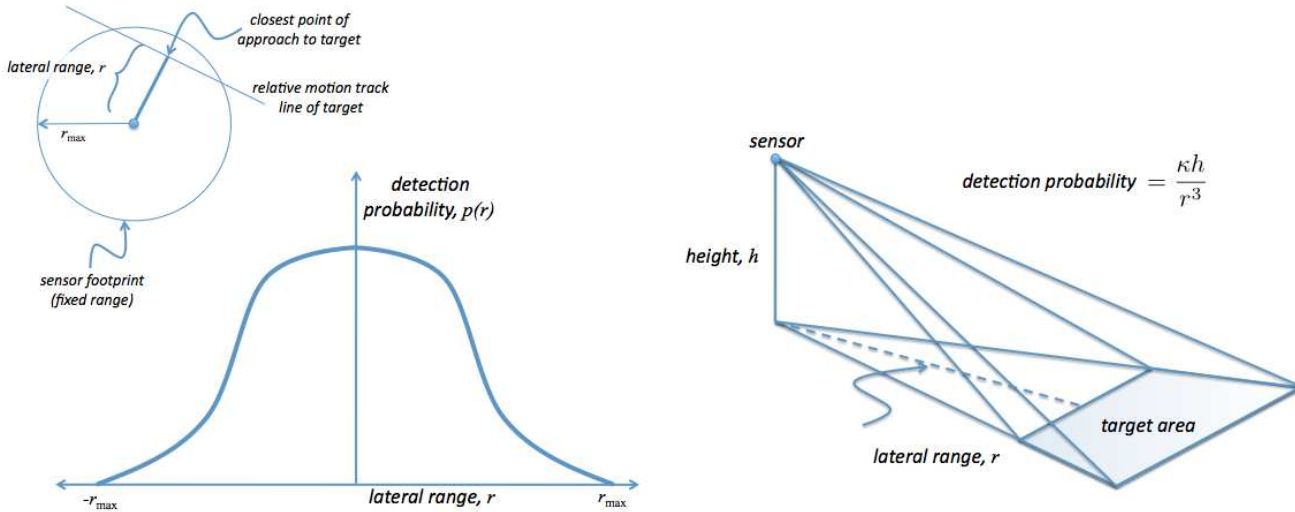
Fig. 5 Illustration of mathematical models for sensor detection characteristics. **Left:** the lateral range curve for a range-dependent sensor such as sonar. **Right:** an inverse cube model for visual (aerial) sightings.

complete problem, and that the minimization of the expected time until detection is an NP-hard one. These results also outline the tight relationship between the two measures of search performance, highlighting their equally prominent examination in the literature.

As a result of the intractability of solutions, much research has since been dedicated to finding near-optimal solutions to variations of the optimal search problem. Relation to the vehicle routing problem (e.g., see Lenstra and Kan (1981); Toth and Vigo (2002)) in operations research offers numerous approximation algorithms, often relying on customized branch-and-bound approaches (Washburn, 1998). These works formulate optimization objective functions with relevance to both types of search performance measures, namely maximal detection probability and minimal (expected) detection time. Further, to address the additional challenge of a moving target, Washburn (1983) proposed an iterative forward-and-backward algorithm to address the case of a Markovian target in a discretized search environment, and Eagle and Yee (1990) provided a branch-and-bound algorithm to find an optimal path for a single moving target. Extensions to these works have led to models for more sophisticated target motion models (e.g., Dambreville and Cadre (2002)). However, the following open problem remains: *How can adversarial behaviors of target(s) (such as the pursuit-evasion results highlighted in Section 2) be incorporated into search theoretic constructs within the constraints of computational tractability?*

More recently, algorithms such as proposed by Sato and Royset (2010) represent the current state-of-the-art in solving these problems computationally. In this work, the optimization model is given by a mixed integer linear program and solved using numerical solvers.

Constraints such as transit costs and search times (that is, time required to inspect a given location) offer operational relevance while maximizing the probability of detection of multiple targets. Such results build upon the foundational works (e.g., Chew (1967)) which define such costs and show that the "most inviting strategy" is optimal, i.e., myopic selection of the next search location that has the highest ratio of probability of target presence to cost of searching and/or transiting to the location is provably best under special assumptions.

Another area in search theory that has seen much activity involves the possibility of "false alarms" or false positive detections. Kadane (1971) first proposed the optimal whereabouts search, in which the searcher seeks a stationary target, or if unable to find it within a fixed search effort budget, specifies its most likely location in the search region. Kadane's work leveraged results from the optimal stopping problem (Ross, 1969; Chew and Milton, 1973), which offer specification of termination criteria for the search process. Such stopping conditions are more relevant once false contacts are introduced to the search problem formulation (Dobbie, 1973). Much of the more recent investigations, such as Kalbaugh (1992) and Hohzaki (2007), studied the effect of these false contacts, which may arise not only from sensor errors but also from the presence of physical objects in the search region distinct from the desired target(s).

Bayesian approaches have also been utilized to address imperfect sensors, recognizing the benefit of adaptive approaches. Assaf and Zamir (1985) utilized prior distributions on locations of multiple objects and constructed theoretical descriptions of posterior distributions in a Bayesian manner from the searcher's ob-

servations. For these classes of probabilistic distributions, the authors further prove that the myopic "most inviting" strategy is optimal, extending their predecessors' results from deterministic to probabilistic domains. This and other contemporary works (e.g., Kimeldorf and Smith (1979)) led to substantial research in the probabilistic modeling communities. However, these theoretical results apply only for special distributions and further rely on the absence of false positive detection errors, both of which are limiting in the case of real world implementations.

Kress et al. (2008) examined how best to allocate search effort, i.e., observations, at discrete locations to detect a stationary target in minimum time. The authors incorporate false positive detections, and their model assumes that a separate effort must be expended to verify the presence or absence of the target. The combined cost of search and verification yields an expression for the expected time until the target's location is correctly identified, for which a myopic rule is shown to be optimal. A remaining open problem includes: *For what broad classes of search problem formulations can a myopic search strategy be proven to be optimal?* The lines of research above bring the probabilistic search field closer to the robotics community, which has been investigating sensor-based search planning in recent years, as discussed in the sequel.

4 Search and Pursuit-Evasion in Robotics

Recently researchers have extended search and pursuit-evasion techniques to allow for implementation on real-world systems. In this section, we will discuss transitional research that bridges the gap between theory and application on robotic systems, and we will highlight field deployments and implementation results. Within the scope of this survey, we limit the discussion to probabilistic search and pursuit-evasion on graphs and in polygonal environments, which have direct impact on robotic applications. We do not consider other forms of pursuit-evasion games (e.g., differential games).

One of the key differences between theory and implementation of search and pursuit-evasion algorithms on mobile robots is that the environment and searchers must be modeled realistically. For instance, graphical representations are limited by the requirement for discretization of both the possible locations and the sensor capabilities. Polygonal representations are somewhat more realistic in that they allow for continuous sensing and movement, but it is often difficult to obtain polygonal maps from sensor data, especially in cluttered environments. A number of discrete and continuous representations have been explored, which lead to various solutions and methods for analyzing these solutions.

In addition, as noted above, many of the techniques applied to mobile robotics can be partitioned into worst-case and average-case techniques based on the assumptions on the target's model and the objective function. Techniques from the probabilistic optimization literature (see Section 3) as well as techniques from the pursuit-evasion literature (see Section 2) have been extended for use on robotic systems. We will now discuss both theoretical and experimental results particularly relevant to robotics research.

4.1 Search in Polygonal Environments

To better model physical environments, robotics researchers have studied formulations of the adversarial search problem on representations other than a graph. Guibas and LaValle et al. extended pursuit-evasion techniques to guarantee detection of an adversarial target in polygonal environments (LaValle et al., 1997; Guibas et al., 1999). The resulting visibility-based pursuit-evasion problem is as follows. Given a bounded continuous environment, pursuers with controlled movements, an evader whose position is unknown to the pursuer, and a visibility sensor that defines an observed subset of the environment based on the position of a pursuer, the task is to find a path for the pursuers that guarantees detecting the evader, regardless of the path it takes. Such a path is said to clear the environment. Some polygonal environments can be cleared with a single pursuer, and others require multiple pursuers. It can be challenging to determine whether or not an environment is clearable with a single pursuer, and small changes to the polygon can actually modify the number of searchers required (see Figure 7). However, Isler et al. (2005) showed that a single pursuer can locate an evader in any simply-connected polygon using a randomized strategy.

One important difference between pursuit-evasion in polygonal environments and pursuit-evasion on the edges of a graph is that polygonal environments can require recontamination (i.e., a previously cleared area must be opened to possible target intrusion during the search) to clear with the minimal number of searchers. In contrast, the edge search game (Parsons, 1976) will never require recontamination to achieve a search strategy with minimal searchers (LaPaugh, 1993). Guibas and LaValle et al. showed that, in fact, a linear number of recontaminations are sometimes necessary for visibility-based pursuit-evasion. More precisely, they show that there exists a sequence of simply-connected free spaces with search number equal to one such that $\Omega(n)$ recontaminations are required for n edges.
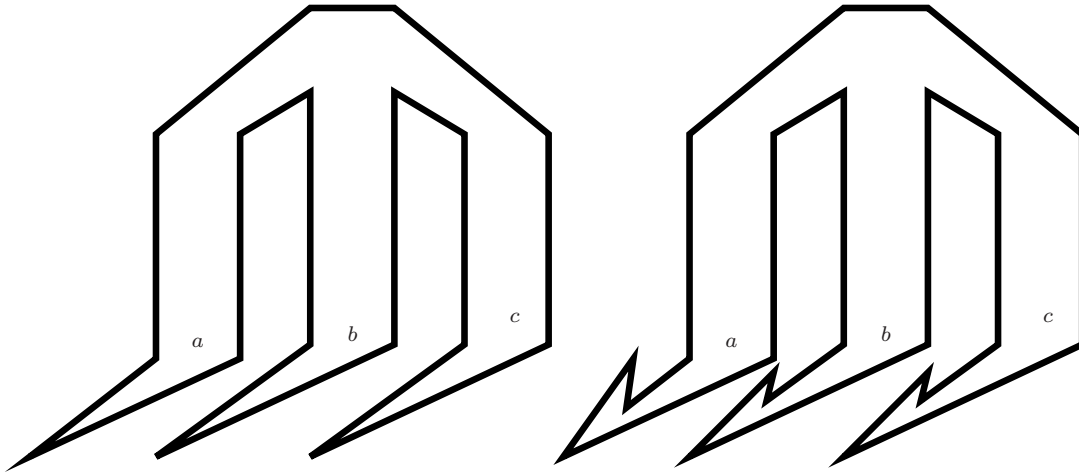
Fig. 7 Visibility-based pursuit-evasion examples. The environment on the left can be cleared by a single searcher with unlimited visibility. Since the evader has unlimited speed, it can avoid the pursuer in the environment on the right. Additional examples can be found in LaValle (2006) Section 12.4. The pursuer can use a randomized strategy to capture the evader in both cases. Compare with Figure 2.

Figure 8 gives an example environment requiring a linear number of recontaminations (Guibas et al., 1999). The peak at the top of the environment must be re-contaminated to continue clearing the legs. The environment can be cleared by a single pursuer with line-of-sight sensing, but the peak will be recontaminated at least $k - 1$ times. The possibility of recontamination increases the difficulty of generating solutions to polygonal search problems, particularly when multiple searchers are required.

To solve the problem of polygonal search with a single searcher, Guibas et al. (1999) introduced the idea of gap edges: edges of a searcher's visibility polygon that border areas that may or may not contain the target. Their approach discretizes polygonal environments into visibility regions in which a robot can move freely without modifying the state of its gap edges (i.e., whether or not the region bordered by the gap edge contains a target). The algorithm then generates and searches an information space defined by the labels of the gap edges combined with the location of the robot. The resulting algorithm is guaranteed to find a solution with a single searcher if one exists, and it has been shown to be efficient in practice (Guibas et al., 1999). The only known worst-case bound on the number of information states generated by this algorithm is exponential in the number of vertices of the polygon. However, it is not clear if this bound is tight.

Park et al. (2001) provided necessary and sufficient conditions for polygonal environments clearable with a single searcher, and they present an $O(n^2)$ algorithm for clearing a polygon with a single pursuer, which im-

proves on the result from Guibas et al. with possibly exponential worst-case running time. An alternative to using deterministic strategies is the randomized strategy given by Isler et al. (2005), which guarantees that a single pursuer can locate an evader in any simply-connected polygon with high probability in time polynomial in the number of vertices.

With multiple searchers, both algorithms by Guibas et al. and Park et al. would generate a potentially exponential number of visibility regions and search the resulting information space. This approach is only tractable for few searchers and small environments. One alternative is to use an iterative visibility-based method, but doing so loses completeness.

The work described above is limited to 2D environments. Lazebnik (2001) provided an initial characterization of the challenges of extending visibility-based techniques to 3D environments. This leads us to an open problem: *Are efficient and complete algorithms possible for clearing 3D environments? What restrictions on geometry and searcher capabilities are necessary for such algorithms to be applicable?*

A similar problem is that of of maintaining visibility to an adversarially moving target in a cluttered environment. This problem was originally introduced by LaValle et al. (1997), and has been extended to the case of limited range (Murrieta-Cid et al., 2007). Bhattacharya and Hutchinson (2010) recently examined the presence of Nash equilibrium for the case of a single pursuer maintaining visibility to an evader with bounded speed. They presented necessary and sufficient conditions for cases where the target can escape the
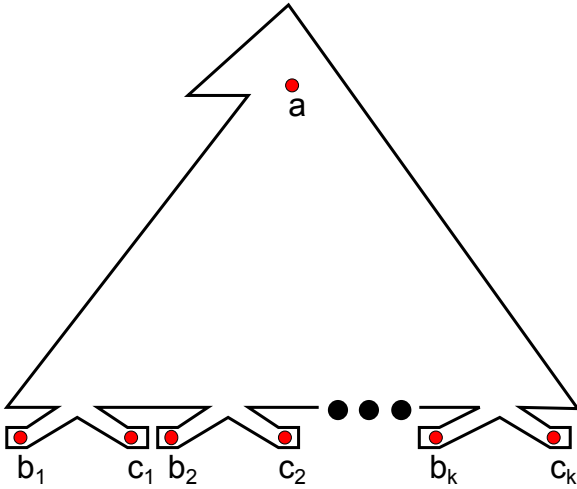
Fig. 8 Environment from (Guibas et al., 1999) that requires a linear number of recontaminations to clear with a single searcher. The peak (labeled a) must be recontaminated as the legs (labeled b and c) are progressively cleared. In contrast with edge search, where recontamination cannot improve the search number, polygonal environments can require recontamination in order to clear with the minimal number of searchers.

pursuer. These methods provide strategies for continuous tracking in surveillance applications. However, few algorithms for maintaining visibility have been implemented on real systems.

4.2 Searching Environments Represented as Graphs

As described in Section 2, a large portion of fundamental work in pursuit-evasion examined the problem of edge search, where the evader resides on the edges of a graph. Edge search does not apply directly to many robotics problems. The possible paths of an evader in many indoor and outdoor environments often cannot be accurately represented as the edges in a graph. In some cases, it is possible to construct a dual graph by replacing the nodes with edges, but these translations do not necessarily yield the same results as the original problem.

Several versions of "node search" appear in the literature. In one formulation, the evader resides in the edges of the graph (hence, despite the name, this is really an edge search problem), and these edges are cleared by trapping (i.e., two searchers occupy the adjacent nodes). Hollinger et al. (2010a) discussed the properties of adversarial search when the evader resides on the nodes, and they show its formal relationship to edge search. Any node search clearing strategy is also an edge search clearing strategy, but the opposite is not true. However, it is NP-hard to determine the minimal searchers required for a clearing strategy for both edge search and node search.

Kolling and Carpin (2010) presented a formulation of the adversarial search problem, which they refer to as GRAPH-CLEAR. The GRAPH-CLEAR problem is represented by a weighted graph where nodes represent arbitrarily shaped areas, and edges represent transitions between these areas. Each node is labeled by the number of searchers required to clear that area, and each edge is labeled by the number of searchers to guard that edge (i.e., prevent an adversary from moving through it). Kolling and Carpin showed that the GRAPH-CLEAR problem of finding the minimal number of pursuers to clear this hybrid graph can be solved efficiently if the topology is a tree, but is NP-hard for general graphs. In addition, they developed a graph cut algorithm to find the minimal number of searchers required for a GRAPH-CLEAR instance on trees. The application of the GRAPH-CLEAR framework to complex environments is a subject of ongoing research.

The traditional formulation of guaranteed search does not restrict the movement of searchers: they are allowed to "teleport" between nodes in the graph without following the edges between them (Parsons, 1976; Megiddo et al., 1988). This assumption enables searchers to clear disjoint parts of the graph without maintaining a route to a starting node. Barrière et al. (2003) introduced the idea of connected search, during which searchers must maintain a connected subgraph of cleared nodes. Connected search guarantees that a path exists to the starting nodes at all times and that searchers are connected by a cleared or "safe" region of the graph. Barrière et al. argued that such a constraint is an important quality for search strategies in the network decontamination domain. Connectedness is also an important characteristic of guaranteed search strategies in robotics applications. Real robots cannot teleport between nodes in the graph because these nodes represent physical locations. Instead, robots must restrict their search paths to those traversable in the environment.

The algorithms described above attempt to minimize the number of searchers necessary to clear a given graph. Researchers have paid far less attention to generating clearing schedules that require minimal time or distance. Minimizing distance and time are of particular interest for robotics applications when a fixed number of robots may be available, but time is limited. Borie et al. (2009) discussed algorithms and complexity results for the minimal time and minimal distance clearing problems. They provided a summary of the known complexity results for the minimum pursuer, minimum distance, and minimum time problems of various types of graphs. Hollinger et al. (2010b) also examined the problem of minimizing distance using a heuristic that

bootstraps on solutions to the adversarial search problem on an underlying spanning tree. Despite these recent efforts, we are left with the following open problems: *A number of minimum time and minimum distance problems remain open on discrete graphs (see Table 1). For such problems with known complexity results, what heuristics and approximations are still possible?*

4.3 Probabilistic Search Methods in Robotics

Both classical pursuit-evasion approaches and more recent graph search algorithms rely on the worst-case assumption of the target's behavior, which precludes modeling the target's motion or incorporating uncertainty into the search plans. An alternative formulation of multi-robot search problems uses a probabilistic approach to model the location of the target or the movement of the searchers.

Many search problems can be formulated as a Markov Decision Process (MDP) if the target's position is fully observed (Eaton and Zadeh, 1962) or a Partially Observable Markov Decision Process (POMDP) if it is unknown. These formulations provide probabilistic representations of the problem, which can reason about uncertainty in measurements and target models. Roy et al. (2005) showed how belief compression can be used to make the POMDP search problem tractable for a single pursuer, but the increased state space makes this approach infeasible as the environment and team size scales up.

A number of approximate sampling-based algorithms are available for solving general POMDPs. A state-of-the-art approach that involves sophisticated sampling of the belief space was proposed by Ong et al. (2010) to allow for mixed observability (e.g., when the locations of the searchers are completely known but the location of the target is unknown). The use of mixed observability extends POMDP capabilities to search problems with up to two robots, but they require a discrete representation of the problem, and Markovian target motion models. The Markovian assumption does not allow any target model that requires a history. Hollinger et al. (2009) also examined a receding horizon approximation algorithm to the search POMDP that achieves performance guarantees by sequentially allocating the search effort, requiring only linear computation in the team size. As in general POMDP methods, they also assume a Markovian motion model, and the computation of their method increases exponentially with the horizon length.

Similar to formulations using partial observability, researchers have applied combinatorial optimization techniques to the coordinated search domain. Lau et al.

(2006) presented a dynamic programming approach for efficiently finding a single non-adversarial target. The approach provides optimal solutions for non-adversarial search, but suffers from scalability issues as the team size and environment size increases. Lau et al. (2005) also proposed a branch and bound approach for finding multiple targets that uses a novel bounding strategy. They tested their approach in simulated environments, where it was able to find optimal solutions with several minutes of running time. Branch and bound techniques have the advantage of finding optimal solutions on termination, as well as providing continually improving suboptimal solutions. Such solutions often suffer from scalability issues, and the development of better bounding techniques is an active area of research.

A large number of techniques assume that the map is known beforehand and does not change. Hespanha et al. used a probabilistic framework to formulate a pursuit-evasion problem on partially known maps. They proposed a one-step greedy algorithm, and they proved that their algorithm generates a one-step Nash equilibrium (Hespanha et al., 2000). The equilibrium guarantees that the searchers and target cannot do better by modifying their strategies using a local search. However, such algorithms do not guarantee global optimality, which leaves the following problem open: *Are there efficient probabilistic search algorithms with approximation guarantees for unknown and dynamic environments?*

4.4 Implementation and Field Results

An increasing number of search and pursuit-evasion algorithms are being implemented on multi-robot systems. A major issue with implementation on such systems is the requirement that the system be robust to failures. This requirement has led to a number of approaches that are decentralized and avoid single points of failure. Vidal et al. (2002) developed decentralized coordination strategies for a UAV/UGV team searching for multiple targets. Their testbed consisted of two Pioneer 2-AT ground vehicles acting cooperatively with a Ymaha R-50 helicopter. All vehicles were equipped with GPS, inertial navigation, onboard computation, a PTZ vision system, and wireless communication. Vidal et al. presented two greedy coordination approaches based on maximizing the probability of detection. They demonstrated empirically that their method is able to locate targets attempting to evade capture. However, their method does not explicitly model the target as adversarial and provides no guarantees in this case.

A number of researchers have applied probabilistic filtering approaches to model the target. Bourgault

Unit-Length Arbitrary-Width Graphs			
	Minimum Pursuer	Minimum Distance	Minimum Time
Paths	P	Pseudo-P	Pseudo-P
Cycles	P	Pseudo-P	Pseudo-P
Stars	NP-Complete, Pseudo-P	NP-Complete	Strongly NP-Complete
Trees	NP-Complete	NP-Complete	Strongly NP-Complete
Two-Vertex Graphs	NP-Complete, Pseudo-P	NP-Complete	Strongly NP-Complete
Series-Parallel Graphs	NP-Complete	NP-Complete	Strongly NP-Complete
Cliques	NP-Complete	NP-Complete	NP-Complete
General Graphs	NP-Complete	NP-Complete	Strongly NP-Complete
Unit-Width Arbitrary-Length Graphs			
	Minimum Pursuer	Minimum Distance	Minimum Time
Paths	P	P	P
Cycles	P	P	P
Stars	P	P	Strongly NP-Complete
Trees	P	(Open)	Strongly NP-Complete
Two-Vertex Graphs	P	P	Strongly NP-Complete
Series-Parallel Graphs	(Open)	(Open)	Strongly NP-Complete
Cliques	P	P	Strongly NP-Complete
General Graphs	NP-Complete	NP-Complete	Strongly NP-Complete

Table 1 Complexity results from (Borie et al., 2009) for various pursuit-evasion problems on different special case graphs. The known hardness results are shown for clearing a discrete graph with multiple pursuers. The evader exists in the edges of the graph, and the pursuers clear edges by traversing them. The pursuers also block the evader's movement by occupying nodes on the graph.

et al. (2003, 2006) examined the problem of locating a potentially moving, non-adversarial target where the target's predicted motion is modeled using a Bayesian filter. In other work, these techniques were extended to multiple targets (Wong et al., 2005). These works applied decentralized data fusion to develop a fully distributed approach. They also examined several candidate utility functions and present an optimal search strategy for a single searcher. While a large portion of this work was validated in simulation, one notable exception is the work of Tisdale et al. (2009), which applied receding-horizon control with a variable horizon to a UAV search problem where the target's position is modeled using recursive Bayesian estimation. Two Sig Rascal airframes were equipped with GPS, wireless communication, onboard computing, and downward looking cameras. Their objective is to maximize the information gain given a new UAV sensor measurement, which yields performance guarantees due to the submodularity of the objective function. Their approach is purely average-case, and provides no guarantees if the target is acting adversarially or if the forward model in the probabilistic filter is violated.

Practical implementation of search algorithms has been demonstrated in the wilderness search and rescue domain. Goodrich et al. (2008) provided an extensive experimental analysis of UAV-enabled search with vision, which includes field trials with search and rescue operators. They utilized a small and light fixed wing UAV equipped with autopilot, GPS, and a video camera on a gimbal mount. The vehicle used a radio transceiver for data communication and an analog transmitter for

video transmission. They presented a number of techniques for mosaics and visual processing to improve how information is viewed by human operators. Their work used a relatively simple contour-based search strategy to guide the UAVs, with the goal of covering the area.

While a number of techniques have been implemented on mobile robots searching for non-adversarial targets, fewer have been implemented for the adversarial search problem. Gerkey et al. (2005) utilized the PARISH algorithm to clear a small academic building with three physical robots equipped with laser scanners. PARISH frames the clearing problem as a parallel optimization, and allows robots to form teams that work together to clear the environment. Team formation and path generation are guided by a heuristic, which makes the algorithms sensitive to choice of heuristic. Vieira et al. (2009) presented experiments with a team of iRobot Create platforms executing adversarial target search using a graph partition method. The robots utilize beacons in the indoor environment to localize themselves, and they execute a team strategy to capture a target moving faster than the searchers. Hollinger et al. (2010a) implemented a building clearing algorithm using a human-robot search team on a single floor of an office building. Their approach utilizes the clearing schedule of an underlying spanning tree as a heuristic to generate clearing schedules with few searchers and fast clearing times. Their search team consisted of a Pioneer differential drive platform with a camera for detection and laser for localization, along with two humans carrying laptops. The humans' laptops communicate through the wireless network with the robot, and

the laptop displays relay waypoint information to the humans.

The three implementations above were on small ground vehicles in indoor environments. Katsilieris et al. (2010) presented a demonstration on two large ground vehicles in an outdoor field with obstacles. The test robots were spherical Rotundus GroundBot vehicles equipped with GPS, wireless communication, and two cameras for 360 degree field of view. The vehicles cleared the outdoor environment of any potentially adversarial target, which the authors refer to as search and secure. The algorithm does not provide guarantees on the number of searchers or the time to complete the search, nor does it account for communication limitations between the robots. However, the robots successfully executed the clearing strategy and were able to guarantee detection of any potential adversary.

The above adversarial search algorithms were designed to operate in 2D planar environments, possibly with obstacles. Kolling et al. (2010) extended heuristic adversarial search techniques to 2.5D environments by utilizing an extended notion of visibility. Their algorithm reduces the 2.5D search problem on a height map to a graph through the use of sampling and calculation of detection sets. Once the graph is constructed, standard 2D adversarial search algorithms can be applied. They demonstrated their algorithm using eight human searchers in an outdoor testing site on the scale of approximately one square kilometer (Kolling et al., 2011). The development of more general techniques for search in higher dimensions remains a topic for future study.

Despite the progress made by these implementations, demonstrations of adversarial search on a team of mobile robots in environments larger than a single floor of a building or a small outdoor area have yet to be presented. Many of the challenges are related to systems engineering for field testing of multi-robot teams. Due to poor reliability and the number of possible failures, few multi-robot systems have been tested on the scale necessary to demonstrate pursuit-evasion in complex environments. In addition, a reliable communication infrastructure is required to guarantee that the plan will be executed without error. Decentralized architectures and methods capable of utilizing new information as it becomes available reduce these requirements, but the development and evaluation of such algorithms remains an active research area.

naturally arising from the increased capabilities of autonomous agents. Practical impact areas include: surveillance, emergency response, and wilderness/ocean rescue. In keeping with its interdisciplinary nature, robotics brings an applied context for revisiting existing theoretical results, as well as inspiring new ones. This paper has highlighted fundamental work in search and pursuit-evasion as found in computer science, operations research, and other communities, and it has provided a survey of a number of recent advancements in robotics.

Many interesting open problems exist in search and pursuit-evasion. The problems of minimizing time and distance to capture in the adversarial domain have seen far less attention than minimizing the number of pursuers. Since computing the number of pursuers is often computationally hard, existing solutions are heuristic in nature. Approximation algorithms with provable performance guarantees are missing, even for many special-case environments. These questions offer a possible bridge with the probabilistic search problem, which traditionally has examined the expected time to find a target. Alternatively, using stochastic optimization methods in the context of max-min pursuit-evasion games (e.g., "attacker-defender" models) can introduce notions of probabilistic adversaries.

With few exceptions, current practical search and pursuit-evasion algorithms have been designed for 2D environments. The extension to 3D and higher dimensions has been studied through theoretical analysis, but the development of algorithms that are efficient for 3D search and pursuit-evasion is still largely an open problem. The extension of graph theoretic search to more general characterizations of visibility, such as 2.5D height maps, provides one promising avenue for future work.

Issues of practical relevance, including bounded speed, constrained mission endurance, refined sensing models, and limited inter-robot communication, continue to be of interest to many robotics research efforts. Such issues must be resolved to enable large-scale testing and fieldwork on robotic search systems. In addition, common software tools and architectures have yet to be developed for search and pursuit-evasion. Libraries of algorithms, environments, and data sets would facilitate ease of comparison for new techniques and provide a resource for researchers entering the field. These and other avenues for future research can be rewarding to all relevant communities, driven by the collective interests in robotic search and pursuit-evasion.

5 Conclusion and Future Research

Search and pursuit-evasion problems have recently become central to many application domains in robotics,

References

Adler, M., Räcke, H., Sivadasan, N., Sohler, C., and Vöcking, B. (2003). Randomized pursuit-evasion in

graphs. *Combinatorics, Probability and Computing*, 12(3):225–244.

Aigner, M. and Fromme, M. (1984). A game of cops and robbers. *Discrete Applied Mathematics*, 8(1):1–12.

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

Alexander, S., Bishop, R., and Ghrist, R. (2009). Capture pursuit games on unbounded domains. *l'Enseignement Mathematique*, 55:103–125.

Alonso, L., Goldstein, A. S., and Reingold, E. M. (1992). Lion and Man: Upper and lower bounds. *INFORMS Journal on Computing*, 4(4):447.

Alspach, B. (2004). Searching and sweeping graphs: A brief survey. *Matematiche*, 59:5–37.

Assaf, D. and Zamir, S. (1985). Optimal sequential search: A bayesian approach. *The Annals of Statistics*, 13(3):1213–1221.

Barrière, L., Fraigniaud, P., Santoro, N., and Thilikos, D. (2003). Searching is not jumping. *Graph-Theoretic Concepts in Computer Science*, 2880:34–45.

Başar, T. and Olsder, G. J. (1999). *Dynamic noncooperative game theory*. Society for Industrial Mathematics.

Benkoski, S. J., Monticino, M. G., and Weisinger, J. R. (1991). A survey of the search theory literature. *Naval Research Logistics*, 38(4):469–494.

Bhadauria, D. and Isler, V. (2011). Capturing an evader in a polygonal environment with obstacles. In *Proc. International Joint Conference on Artificial Intelligence*.

Bhattacharya, S. and Hutchinson, S. (2010). On the existence of nash equilibrium for a visibility based pursuit evasion game. *International Journal of Robotics Research*, 29(7):831–839.

Bienstock, D. and Seymour, P. D. (1991). Monotonicity in graph searching. *Journal of Algorithms*, 12(2):239–245.

Bopardikar, S. D., Bullo, F., and Hespanha, J. P. (2007). Sensing limitations in the Lion and Man problem. In *Proc. American Control Conference*, pages 5958–5963.

Borie, R., Tovey, C., and Koenig, S. (2009). Algorithms and complexity results for pursuit-evasion problems. In *Proc. International Joint Conference on Artificial Intelligence*, pages 59–66.

Bourgault, F., Furukawa, T., and Durrant-Whyte, H. F. (2003). Coordinated decentralized search for a lost target in a bayesian world. In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 48–53.

Bourgault, F., Furukawa, T., and Durrant-Whyte, H. F. (2006). Optimal search for a lost target in a bayesian world. *Field and Service Robotics*, 24:209–222.

Chew, M. C. (1967). A sequential search procedure. *The Annals of Mathematical Statistics*, 38(2):494–502.

Chew, M. C. and Milton, C. (1973). Optimal stopping in a discrete search problem. *Operations Research*, 21(3):741–747.

Dambreville, F. and Cadre, J. P. L. (2002). Search game for a moving target with dynamically generated informations. In *Proc. International Conference on Information Fusion*, pages 243–250.

Dobbie, J. M. (1968). A survey of search theory. *Operations Research*, 16(3):525–537.

Dobbie, J. M. (1973). Some search problems with false contacts. *Operations Research*, 21(4):907–925.

Eagle, J. N. and Yee, J. R. (1990). An optimal branch-and-bound procedure for the constrained path, moving target search problem. *Operations Research*, 38(1):110–114.

Eaton, J. and Zadeh, L. (1962). Optimal pursuit strategies in discrete-state probabilistic systems. *Journal of Basic Engineering*, 84:23–28.

Fomin, F. V. and Thilikos, D. M. (2008). An annotated bibliography on guaranteed graph searching. *Theoretical Computer Science*, 399(3):236–245.

Gerkey, B., Thrun, S., and Gordon, G. (2005). Parallel stochastic hill-climbing with small teams. In *Proc. International NRL Workshop on Multi-Robot Systems*.

Goldstein, A. S. and Reingold, E. M. (1995). The complexity of pursuit on a graph. *Theoretical Computer Science*, 143(1):93–112.

Goodrich, M., Morse, B., Gerhardt, D., Cooper, J., Quigley, M., Adams, J., and Humphrey, C. (2008). Supporting wilderness search using a camera-equipped UAV. *Journal of Field Robotics*, 25(1–2):89–110.

Guibas, L., Latombe, J., LaValle, S., Lin, D., and Motwani, R. (1999). Visibility-based pursuit-evasion in a polygonal environment. *International Journal of Computational Geometry and Applications*, 9(5):471–494.

Hespanha, J., Prandini, M., and Sastry, S. (2000). Probabilistic pursuit-evasion games: A one-step nash approach. In *Proc. IEEE Conference on Decision and Control*, pages 2272–2277.

Hohzaki, R. (2007). Discrete search allocation game with false contacts. *Naval Research Logistics*, 54(1):46–58.

Hollinger, G., Kehagias, A., and Singh, S. (2010a). GSST: Anytime guaranteed search. *Autonomous*

Robots, 29(1):99–118.

Hollinger, G., Kehagias, A., and Singh, S. (2010b). Improving the efficiency of clearing with multi-agent teams. *International Journal of Robotics Research*, 29(8):1088–1105.

Hollinger, G., Singh, S., Djugash, J., and Kehagias, A. (2009). Efficient multi-robot search for a moving target. *International Journal of Robotics Research*, 28(2):201–219.

Isler, V., Kannan, S., and Khanna, S. (2005). Randomized pursuit-evasion in a polygonal environment. *IEEE Transactions on Robotics*, 21(5):875–884.

Isler, V., Kannan, S., and Khanna, S. (2006). Randomized pursuit-evasion with local visibility. *SIAM Journal on Discrete Mathematics*, 1(20):26–41.

Isler, V. and Karnad, N. (2008). The role of information in the cop-robber game. *Theoretical Computer Science*, 3(399):179–190. Special Issue on Graph Searching.

Jankovic, V. (1978). About a man and lions. *Matematicki Vesnik*, 2:359–361.

Kadane, J. (1971). Optimal whereabouts search. *Operations Research*, 19(4):894–904.

Kalbaugh, D. (1992). Optimal search among false contacts. *SIAM Journal on Applied Mathematics*, 52(6):1722–1750.

Karnad, N. and Isler, V. (2008). Bearing-only pursuit. In *Proc. IEEE International Conference on Robotics and Automation*, pages 2665–2670.

Katsilieris, F., Lindhé, M., Dimarogonas, D., Ögren, P., and Johansson, K. (2010). Demonstration of multi-robot search and secure. In *Proc. ICRA Workshop on Search and Pursuit/Evasion*.

Kimeldorf, G. and Smith, F. (1979). Binomial searching for a random number of multinomially hidden objects. *Management Science*, 25(11):1115–1126.

Kolling, A. and Carpin, S. (2010). Pursuit-evasion on trees by robot teams. *IEEE Transaction on Robotics*, 26(1):32–47.

Kolling, A., Kleiner, A., Lewis, M., and Sycara, K. (2010). Pursuit-evasion in 2.5D based on team-visibility. In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4610–4616.

Kolling, A., Kleiner, A., Lewis, M., and Sycara, K. (2011). Computing and executing strategies for moving target search. In *Proc. IEEE International Conference on Robotics and Automation*, pages 4246–423.

Koopman, B. O. (1956a). The theory of search, part i. kinematic bases. *Operations Research*, 4(5):324–346.

Koopman, B. O. (1956b). The theory of search, part ii. target detection. *Operations Research*, 4(5):503–531.

Koopman, B. O. (1957). The theory of search, part iii. the optimum distribution of searching effort. *Operations Research*, 5(5):613–626.

Koopman, B. O. (1979). Search and its optimization. *The American Mathematical Monthly*, 86(7):527–540.

Kopparty, S. and Ravishankar, C. V. (2005). A framework for pursuit evasion games in Rn. *Information Processing Letters*, 96(3):114–122.

Kress, M., Lin, K. Y., and Szechtman, R. (2008). Optimal discrete search with imperfect specificity. *Mathematical methods of operations research*, 68(3):539–549.

LaPaugh, A. S. (1993). Recontamination does not help to search a graph. *Journal of the Association for Computing Machinery*, 40(2):224–245.

Lau, H., Huang, S., and Dissanayake, G. (2005). Optimal search for multiple targets in a built environment. In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3740–3745.

Lau, H., Huang, S., and Dissanayake, G. (2006). Probabilistic search for a moving target in an indoor environment. In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3393–3398.

LaValle, S., Lin, D., Guibas, L., Latombe, J., and Motwani, R. (1997). Finding an unpredictable target in a workspace with obstacles. In *Proc. IEEE International Conference on Robotics and Automation*, pages 737–742.

LaValle, S. M. (2006). *Planning Algorithms*. Cambridge University Press.

Lazebnik, S. (2001). Visibility-based pursuit-evasion in three-dimensional environments. Technical report, CVR-TR-2001-01, Beckman Institute, University of Illinois at Urbana-Champaign.

Lenstra, J. K. and Kan, A. H. G. R. (1981). Complexity of vehicle routing and scheduling problems. *Networks*, 11(2):221–227.

Littlewood, J. E. (1953). *A Mathematician's Miscellany*. Methuen & Co. Ltd., London.

McCue, B. (1990). *U-Boats in the Bay of Biscay: an essay in Operations Analysis*. National Defense University Press.

Megiddo, N., Hakimi, S., Garey, M., Johnson, D., and Papadimitriou, C. (1988). The complexity of searching a graph. *Journal of the Association of Computing Machinery*, 35(1):18–44.

Murrieta-Cid, R., Muppirala, T., Sarmiento, A., Bhattacharya, S., and Hutchinson, S. (2007). Surveillance strategies for a pursuer with finite sensor range. *International Journal of Robotics Research*, 26(3):233–253.

Nowakowski, R. and Winkler, P. (1983). Vertex-to-vertex pursuit in a graph. *Discrete Mathematics*, 43(2–3):235–239.

Ong, S. C. W., Png, S. W., Hsu, D., and Lee, W. S. (2010). Planning under uncertainty for robotic tasks with mixed observability. *International Journal of Robotics Research*, 29(8):1053–1068.

Park, S.-M., Lee, J.-H., and Chwa, K.-Y. (2001). Visibility-based pursuit-evasion in a polygonal region by a searcher. In *Proc. International Colloquium on Automata, Languages and Programming*, pages 456–468.

Parsons, T. (1976). Pursuit-evasion in a graph. In Alavi, Y. and Lick, D., editors, *Theory and Applications of Graphs*, pages 426–441. Springer, Berlin/Heidelberg.

Ross, S. M. (1969). A problem in optimal search and stop. *Operations Research*, 17(6):984–992.

Roy, N., Gordon, G., and Thrun, S. (2005). Finding approximate POMDP solutions through belief compression. *Journal of Artificial Intelligence Research*, 23:1–40.

Sato, H. and Royset, J. O. (2010). Path optimization for the resource-constrained searcher. *Naval Research Logistics*, 57(5):422–440.

Sgall, J. (2001). Solution of David Gale's lion and man problem. *Theoretical Computer Science*, 259(1–2):663–670.

Stone, L. D. (1989a). *Theory of Optimal Search*. Academic Press, 2nd edition.

Stone, L. D. (1989b). What's happened in search theory since the 1975 Lanchester Prize? *Operations Research*, 37(3):501–506.

Tisdale, J., Kim, Z., and Hedrick, J. K. (2009). Autonomous path planning and estimation using UAVs. *IEEE Robotics and Automation Magazine*, 16(2):35–42.

Toth, P. and Vigo, D. (2002). *The Vehicle Routing Problem*. Society for Industrial Mathematics.

Trummel, K. E. and Weisinger, J. R. (1986). The complexity of the optimal searcher path problem. *Operations Research*, 34(2):324–327.

Vidal, R., Shakernia, O., Kim, H. J., Shim, D. H., and Sastry, S. (2002). Probabilistic pursuit-evasion games: Theory, implementation, and experimental evaluation. *IEEE Transactions on Robotics and Automation*, 18(5):662–669.

Vieira, M., Govindan, R., and Sukhatme, G. S. (2009). Scalable and practical pursuit-evasion with networked robots. *Journal of Intelligent Service Robotics*, 2(4):247–263.

Wagner, D. H. (1999). *Naval Operations Analysis*. United States Naval Institute.

Washburn, A. R. (1983). Search for a moving target: The FAB algorithm. *Operations Research*, 31(4):739–751.

Washburn, A. R. (1998). Branch and bound methods for a search problem. *Naval Research Logistics*, 45(3):243–257.

Washburn, A. R. (2002). *Search and Detection*. Topics in Operations Research Series. INFORMS, 4th edition.

Wegener, I. (1985). Optimal search with positive switch cost is NP-hard. *Information Processing Letters*, 21(1):49–52.

Wong, E.-M., Bourgault, F., and Furukawa, T. (2005). Multi-vehicle bayesian search for multiple lost targets. In *Proc. IEEE International Conference on Robotics and Automation*, pages 3169–3174.