Robotica (2006) volume 24, pp. 759–773. © 2006 Cambridge University Press
doi:10.1017/S0263574706002931 Printed in the United Kingdom

New visibility-based path-planning approach for covert robotic navigation

Mohamed S. Marzouqi* and Ray A. Jarvis

Intelligent Robotics Research Centre, Australian Research Council Centre for Perceptive and Intelligent Machines in Complex Environments, Monash University, VIC 3800, Australia

(Received in Final Form: June 17, 2006. First published online: August 8, 2006)

Summary
A new promising approach for visibility-sensitive path-planning problems is presented. The paper focuses on covert navigation where a mobile robot needs to plan a stealthy path to approach a designated destination in a cluttered environment. The aim is to minimize the robot's exposure to hostile sentries within the same environment. The approach can be adapted to work with different levels of initial knowledge the robot may have about both the environment map and the sentries' locations. The approach depends on estimating a cost value at each free-space location that presents the risk of being seen by any sentry. Based on the distance transform algorithm methodology, the minimum visibility–distance cost to a goal is calculated at each cell in the grid-based environment map. Moving along the steepest descent trajectory from any starting point generates an optimal covert path to a goal. The approach has been evaluated with both simulated and physical experiments. A number of test cases are presented. In each case, a path with considerable covertness, compared to a short path to the same destination, is generated. In addition to covert navigation, the approach is introduced briefly as a potential solution for other visibility-based path-planning problems.

KEYWORDS: Visibility-based path planning; Stealth navigation; Mobile robot; Distance transform (DT).

1. Introduction

Covert (or stealthy) navigation is defined here as minimizing self-exposure to movable or stationary observers, possibly hostile, within the same obstacle-strewn environment. Covert robotics systems can be useful in various applications, especially where safety and security are required. The need for mobile robots to be involved in covert missions is obviously due to the dangerous nature of these missions that can threaten human life. Policing and military fields are examples where covert robots can be used for applications such as surveillance, spying, and crime prevention missions. For instance, a robot might be used to covertly gather information from an enemy environment (e.g., a hostage area). In a different application, a domestic service robot may simply want to stay unobtrusive while operating in its environment. Game AI is yet another area that may benefit from the covert robotics field's algorithms.

Different covert applications may require distinct solutions. In general, to accomplish any real-life covert mission successfully, a robot will need to have one or more different covert navigation abilities. Examples of such abilities are: navigating from a point to another without being observed by possible observers, hiding from a pursuer, tracking or approaching a target covertly, finding a safe hidden spot as a base to monitor an area, searching/exploring a hostile environment covertly, and communicating or sharing knowledge with other cooperative team members to achieve maximum covertness.

Obviously, path planning is a crucial part of these abilities. Most of the current and previous research attention for planning an optimal path has been on minimizing distance, time, or power.[8–10] On the other hand, the survey we conducted on the field of covert path planning (Section 2) has shown a weakness in terms of the efficiency of the proposed solutions.

A new visibility-based path-planning approach is introduced here that allows a mobile robot to plan a path with reasonable covertness to a designated destination. The new approach, which we have called the dark path (DP) algorithm, can deal with complex structured environments as well as any number of possible sentries. Moreover, it works with different levels of initial knowledge (i.e., fully known, partially known, or unknown) the robot may have about both the operating environment and the sentries' locations. The available information is used to achieve the maximum possible covertness to facilitate mission success.

The DP algorithm is based on estimating an observability risk cost at each nonobstacle (free or unknown) cell in the uniform grid-based environment map, as can be judged with current knowledge. It is considered an extension to the distance transform (DT) algorithm, a known short collision-free path-planning algorithm,[19] where the minimum visibility–distance cost to the goal is calculated at each free-space cell. Accordingly, moving down a steepest descent trajectory in this cost field from any starting point would generate a path with considerable covertness compared to a short but perhaps risky path, between the same starting and destination points.

* Corresponding author.
E-mail: mohamed.marzouqi@eng.monash.edu.au

https://doi.org/10.1017/S0263574706002931 Published online by Cambridge University Press

The organization of this paper is as follows. In Section 2, a survey on the field of covert path planning is presented. The definition and assumptions of the problem are presented in Section 3. The DP algorithm is explained in detail in Section 4. The DT algorithm is briefly described in the same section. Each of the following four sections (Sections 5–8) describes, with the aid of simulated results, how the DP algorithm can be adapted to solve covert path-planning problems given different levels of initial knowledge. In addition to the simulated experiments, the approach has been tested on a physical robot. Some results are demonstrated in Section 9. The approach's computation time is presented in Section 10. The DP algorithm as a solution for other kind of visibility-based path-planning problems is discussed in Section 11. The conclusion and our direction for future research are given in Section 12.

2. Literature review

The research on covert path planning has not produced many notable outcomes (at least in the public domain literature). Our survey can be divided into two parts: indirectly and directly related works.

2.1. Indirectly related works

The research works in this part have no actual covert path planning; instead, they are attempts to add covertness using different strategies. One example is to plan a path to a dark spot in an environment to hide from expected observers. It has been used in a number of research problems for the purpose of monitoring an area covertly.[11,12] In such research, the robot is equipped with a light-intensity sensor to search for the darkest area where the robot can hide, or at least be less noticeable. The success of this strategy depends heavily on the environment's lighting conditions.

Another strategy to achieve covertness during navigation is to minimize the size of the robot. Insect robotics[13,14] is an obvious example that can be used for such a purpose. Dependence upon the robot's size for covertness is realistic, since an insect robot is either difficult to notice or difficult to be recognized as a robot if observed. On the other hand, a robot's capability decreases with its size. Insect robots are mainly used for spying but not for heavy-duty jobs (e.g., carrying an object) that might be required.

Other examples are pursuit–evasion and hide-and-seek games, which are known problems in autonomous robots field. The aim is to guide one or a team of robots to search for or track one or a group of robots.[15–18] To the best of our knowledge, the work done in this area has only focused on developing the pursuer/seeker side, while the evader/hider is controlled manually or has a random behavior to escape the opponent's observation. One simple strategy depends on finding the path to the nearest position outside the pursuer's current field of view.

2.2. Directly related works

More related, but limited, research works are presented here where a robot plans a path that minimizes its exposure to observers within the environment. These works can be divided into two categories: covert path planning between two predefined points, and covert target tracking/approaching.

2.2.1. Covert path planning to a specified goal. Two approaches are presented elsewhere[20,21] for covert navigation in initially known, natural-terrain environments that are assumed to be free of obstacles. A robot vehicle makes use of the terrain elevation to maintain its covertness from observers at known positions. In reference [20], areas visible to the observers are treated as pseudo-obstacles. A completely covert path to a specified goal is generated accordingly. In reference [21], the observers are assumed to be moving unpredictably. A covert path to a specified destination is found incrementally. At each time interval (within which an observer's motion trajectory is assumed to be fixed), the robot finds the unobserved region connected to its location. Within this region, it navigates to a subgoal that is the nearest to the final goal. However, both of these approaches do not address the case when a completely covert path is not available or the environment is unknown.

In reference [22], a solution is presented for planning a covert path to a goal in an initially unknown environment (with occluding obstacles) and one known stationary observer. The potential field algorithm is used for planning the path. As the environment knowledge is accumulated, "waypoints" are extracted continuously from areas occluded from the observer. When a suitable waypoint is found, the goal force is disabled. The robot is pulled toward the nearest waypoint that lies within a predetermined range of angles between the robot and the goal. If no waypoints fill this criterion then the robot is pulled toward the goal. The same approach is extended elsewhere[23] to dynamic environments. In this case, if a moving obstacle is offering the next waypoint, the robot remains within its occluding area while it moves toward the goal.

As stated elsewhere,[22] the weakness of this approach is inherited from the potential field algorithm, where the robot can easily get entrapped in local minima. Hence, only simple structured environments were considered to test the approach. Moreover, the optimality of the covert path is not guaranteed as the potential field algorithm is a local planner. Another weakness is the need to manually set the range of angles where the waypoints should be considered to lie (depending on the environment's layout). The DP algorithm overcomes these drawbacks as will be described later.

2.2.2. Approaching/tracking a target covertly. Other related works consider problems that require covert behavior when dealing with a hostile target. In reference [22], the approximate location of the target is known initially. Part of the presented problem is to reach the border of the target's view field for spying purposes. Their approach deals with inaccuracies involved in the target's location by cautiously exposing the robot to the target at the terrain-occluding contour. Another covert behavior is presented elsewhere.[26] The aim is to track an unpredictable moving target in a stealthy manner. The proposed solution is to maintain the robot's position inside, but close, to the boundary of the target's field of view. This would allow the robot to quickly escape the target's view if it detects the risk of being observed.

https://doi.org/10.1017/S0263574706002931 Published online by Cambridge University Press

In contrast, some experiments are presented in Section 5 and 6 that introduce the new approach as a solution for approaching and capturing a target covertly, without giving it a chance to escape or take any offensive action.

3. Problem definition

3.1. Problem assumptions

Covert path planning is not a problem that could be solved by a single solution. Different solutions might be required for different assumptions about the environment, the sentries, or the robot's capabilities. To narrow the generality of the problem, the following assumptions have been made.

The operating environment is represented in two-dimensional (2-D) space such that each of its positions cells is either a free-space element or occupied by an obstacle that no agent (i.e., a covert robot or a sentry) can penetrate. The free spaces are assumed to be at the same elevation. Obstacles within the environment are the only means of maintaining covertness. They are assumed opaque and higher than the sight level of all agents. The environment is assumed to be stationary which means that neither the obstacles nor the free spaces change. It can be indoor or outdoor but with specified real or virtual borders to limit the search space for a path.

The agents can only travel within the 2-D free space of the environment. Sentries can stand still (e.g., fixed surveillance camera) or move around (e.g., humans). The covert robot is assumed to have an omnidirectional vision system and a line-of-sight that spans its operating environment. This ability is important as it allows the robot to immediately maintain its covertness once it gets in the view field of an unpredicted sentry. A panoramic vision system is one solution that has been used in the physical experiment as shown in Section 9. The sentries' visual capabilities are assumed to be unknown; therefore, they are considered to be similar to the robot's vision in the worst case. Moreover, all agents have the ability to recognize other agents within their view fields.

In some cases, a completely covert path that a robot can traverse without being observed by any sentry may not be available. Reaching the specified destination is assumed to have a higher priority than only avoiding exposure. Therefore, a path with minimum exposure should be followed, as would be judged with the current knowledge. The time factor is important for high-level covert navigation problems where the robot may use strategies that allow it to change its speed or completely stop for a period of time along its path to avoid exposure. However, this level of complexity is beyond the scope of the current study and is therefore left for future research. The emphasis here is on planning a covert path that the robot would traverse at constant speed from the starting point to the destination.

It should be noted that the aim of this research is *not* to find a solution to a particular application, but to provide a general methodology that can be adapted to deal with various applications each with specific requirements. Simple assumptions are made here to concentrate on presenting the main idea. The presented approach can be extended to deal with other assumptions (which will not be covered here) such as the robot's limited vision, inconsistent ground elevation, and the existence of transparent or short obstacles that cannot be used for hiding.

3.2. Covert path definition

The fundamental difficulty in planning a covert path to a designated goal is to combine shortness with covertness. Maximum covertness might be required in some applications at the cost of distance. An optimal covert path in this case is the shortest path among those with minimum exposure to sentries anywhere in the free space of the environment. In other applications, shortness might be as important as covertness (e.g., the robot needs to reach a goal covertly within a limited time). An optimal path in this case should reflect a balance between these two factors.

4. Covert path-planning approach

Our approach toward the presented problem is the DP algorithm. It is a variation of the DT algorithm, which is briefly described in the next subsection. The new approach has shown its ability to produce acceptable covert paths for different cases of initial knowledge about the environment and the sentries. A detailed description of the new approach is presented in Section 4.2.

4.1. The DT algorithm

The DT algorithm is a valuable global path-planning algorithm[19] that can be extended to handle incremental world modeling, and such ideas as covert navigation. The core idea of this algorithm was presented elsewhere[24] as a numerical potential field method called the Wavefront algorithm. Next, the DT algorithm is explained briefly.

Basically, this algorithm finds the shortest collision-free path between two points in a known stationary environment. The environment is represented as a uniform grid-based map of either free space or obstacle cells. Initially, a *distance map* is created with the same size as the environment map. All the cells are assigned a very high value (e.g., the total number of cells) except the goal cell that is given a distance cost of zero. A distance transformation method (e.g., the forward and backward raster scan[28]) is used to spread the distance costs from the goal cell toward the connected free space. The distance cost at each cell (except the goal) is calculated using (1), where $d\,\mathrm{Cost}()$ represents a recursive function. The final distance cost at each cell is the minimum cost propagated from its eight neighboring cells n. A propagated cost from a neighbor cell is its distance cost plus the Euclidean distance (ED) between the two cells' exact locations. The ED is basically 1 to the orthogonal neighbors and $\sqrt{2}$ to the diagonal ones. Accordingly, every free-space cell in the distance map gets a value that represents the minimum number of cell-steps to the goal. Cells that are not accessible from the goal will remain with the initially assigned value.

$$d\,\mathrm{Cost}(c) = \min_{i=1}^{8}[d\,\mathrm{Cost}(n_i) + \mathrm{ED}_{n_i,c}] \tag{1}$$

At this point, the optimal path can be found, if available, from any starting point to the goal by moving along the steepest descent path (i.e., moving to the next neighbor cell

https://doi.org/10.1017/S0263574706002931 Published online by Cambridge University Press

with the lowest distance cost). The DT algorithm can work as well with multiple goals (e.g., nearest exit door problem), where all the cells representing optional goals are initially assigned a cost of zero. As a result, a path can be generated from any starting point to its nearest goal.

With a simple modification, the DT algorithm can be applied to the case of initially unknown environments, but indeed with no guarantee of finding the shortest path (since environmental mapping is acquired incrementally as the robot moves). An assumption should be made that all the unknown areas are free spaces unless discovered otherwise.[32] The algorithm is re-applied to update the path only when an obstacle is detected along the previously planned path. For this case, an efficient approach to repair the path in real time is the D* algorithm.[33]

4.2. The DP algorithm

An important advantage offered by the DT algorithm is its ability to accept additional cost factors other than the distance. In reference [27], the DT algorithm was used to find a path with safe distance to obstacles, to minimize collision risk. Two cost-propagation processes were conducted. The first one is propagated from the obstacles to assign each cell a value that is inversely proportional to the distance to the nearest obstacle. A second propagation is from the goal that combines the distance cost to the goal with the previously assigned costs at each cell. The generated path from any starting point has both properties of short distance to the goal and safe distance from obstacles. This advantage has motivated the idea of the DP algorithm (the main contribution of this research) as a promising solution for covert navigation problems.[1] It is also considered as a potential solution for other visibility-based path-planning problems as will be discussed in Section 9.

A visibility risk cost (or *visibility cost* for short) is introduced here as a second cost to generate a path with covertness properties. The difference between the distance cost in the DT algorithm and the visibility cost in the DP algorithm should be understood. The distance cost is mainly associated with the robot motion, i.e., the traverse from one cell to another. Therefore, the total distance cost during a navigation task is the length of the traveled distance, regardless of the spent time. Unlike the distance cost, the visibility cost is associated with the observability time as well as the robot's location. As long as the robot is visible to a sentry, it should pay that cost continuously, whether or not it is moving. Therefore, the total visibility cost during a mission is the sum of the visibility costs paid at each moment of the mission, where the visibility cost at any moment is the cost associated with the robot's location at that time.

As inherited from the DT algorithm, the DP algorithm works with a grid-based bounded environment. The DP algorithm consists of two sequential procedures: estimating the visibility map and generating the covert path.

The visibility map is a stand-alone 2-D array that is generated by measuring and assigning a visibility cost for every nonobstacle (i.e., free space or unknown) cell in the environment map. The estimated value at any location represents the risk level of being exposed to one or more sentries. For instance, in the case of a known environment and known sentries, the visibility cost at a cell can be simply 1 if observed by any sentry, or 0 otherwise.

However, visibility map estimation is not a straightforward process for the following two reasons. First, the level of covertness in the final path depends on the precision of the estimated costs. Different applications would have different visibility risk factors that should be taken into account, other than simply being/not being observed by a sentry. Second, the estimation method may differ from case to case depending on the robot's current knowledge. Both the environment and the sentries' locations can be initially known, partially known, or unknown. With uncertainties involved, an estimation method should adapt to the available information as well as the incremented knowledge during navigation. Visibility map estimation is described in detail in the following sections.

The next step is to create the covert path, given both the generated visibility map and the current environment map. A *visibility–distance map* is created where the visibility cost values and the distance-to-goal costs are integrated and propagated from the specified goal through its connected free space. Similar to the DT algorithm, each free-space cell c is assigned a value equal to the least value amongst the eight neighbors' values added to each of their ED, but with the difference here of adding the visibility cost of c. Theoretically, it is more accurate to add the average visibility cost of c and the neighbor cell that has propagated its cost to c. For simplicity, it is set here as the cost of c since adjacent cells would often have very similar costs. Calculating the minimum visibility–distance cost $vd\text{Cost}()$ at each cell is shown as

$$vd\text{Cost}(c) = \min_{i=1}^{8}[\text{Cost}(n_i) + \text{ED}_{n_i,c}] + \alpha V(c) \qquad (2)$$

where $V(c)$ is the visibility cost function that estimates the visibility risk at c. The weight α is a nonnegative real number that determines the required level of covertness in the final path.

The final cost at a cell can be defined as the visibility cost of that cell adjusted by both the visibility risk and the distance costs to the goal. A covert path from any point to the goal is generated by moving along the steepest descent path. In case of multigoals, the chosen goal from any starting point is the one providing the most covert path (compared to the covert paths to the other goals). The chosen path is represented as

$$P(s) = \min_{p \in K}\left(L(p) + \alpha \times \sum_{c_i \in p} V(c_i)\right) \qquad (3)$$

where $P(s)$ represents the optimal path among all possible paths K from a starting cell s to the goal. The optimal path is the one with minimum total visibility–distance cost. The cost of path $p \in K$ is theoretically calculated by adding the length of the path $L(p)$ to the summation of its cells' visibility costs $V(c)$ weighted by α.

The generated path characteristic can be described as a balance between shortness and covertness that can be controlled by adjusting α. The path covertness would reach a maximum limit that would not increase by further weight increment; which is obviously due to the limited number of

https://doi.org/10.1017/S0263574706002931 Published online by Cambridge University Press

possible paths to the goal. At this point, the path is described as the *most covert path*. On the other hand, the shortest path can be generated by setting α to 0. So far, α is assumed to be set manually. It is not clear how the path balance can be adjusted autonomously. Different factors would affect the balance decision when sentries' risk or the priority to reach the goal change quickly during the path. Regardless of the fact that these factors are hard to sense, their integration would form a hard decision problem that would require human-level intelligence. A halfway solution is to allow the robot to set α to a selected weight based on the kind of risk it detects using its sensing capabilities.

An important feature inherited from the DT algorithm, as a global path planner, is that a generated covert path is free from local minima (which is assured by the continuous increment of the visibility–distance cost as it propagates away from the goal). Another inherited feature is the path optimality, which is important for creating a path with the highest possible covertness. This is vital in most covert missions where a mission may fail if the robot is detected. Basically, the optimality depends on the accuracy of the costs calculated at each cell. While distance costs are measurable and no two people would argue on its accuracy, the visibility-cost accuracy on the other hand is arguable (as strategies may differ especially with uncertainties involved). Hence, the accuracies of both the cost estimator and the path generator should be kept separate. Given that we accept the estimated visibility costs to be accurate, the covert path (which is simply created by moving along the steepest descent path) would always be optimal as it is generated efficiently according to the estimation.

In the next four sections, the visibility map is generated for different levels of complexity hierarchy; starting with complete initial knowledge about the environment map and sentries' locations, and ending with no knowledge of these elements. Simulated experiments are presented for each case.

5. Known sentries in known environment

5.1. Visibility map estimation

In the case where both the environment and the sentries' locations are initially known and stationary,[1] the most covert path (which may not be completely covert) is guaranteed given the accuracy of the estimated costs. Estimating accurate costs can be done by determining the visibility risk factors involved in an application. Two risk factors are taken into account here to find a visibility cost at a cell: (1) the number of sentries observing that cell, and (2) the distance to each observing sentry.

These factors usually satisfy a majority of real-life situations. The resultant behavior of the robot is to be observed by the least number of sentries from the farthest possible distance at any time when a completely covert path is not available. This covert behavior is important where, in reality, following a path that might be exposed to one sentry should have a less risk of being detected than a path possibly exposed to two or more sentries. One reason is that sentries might have a less than 360° vision; hence, the possibility of being in the field of view of any sentry will be proportional to their number.

Moreover, being exposed to a sentry at a far distance can minimize the risk of being observed given that sentries might have a limited line-of-sight vision (e.g., in a foggy environment). Another reason is that object detection is assumed to be inversely proportional to open, line-of-sight distance (since this corresponds to the robot's size appearance). In addition, if the robot is detected at a far distance then it would have a better chance to escape or hide.

The resultant visibility map is calculated as follows. At each free-space cell, the visibility risk cost to each sentry observing that cell is calculated. This value is inversely proportional to the cell–sentry separation distance. The final visibility cost at a cell would be the sum of all visibility costs calculated for each of its observing sentries. The following equation represents the visibility cost function.

$$V(c) = \sum_{i=0}^{\text{os}} 1 - \frac{d_i}{d_{\max}}, \qquad d_i \le d_{\max} \tag{4}$$

where $V(c)$ is the visibility cost function calculated at cell c, os is the number of sentries observing c, d_i is the distance to the observing sentry i, and $d_{\max}$ is the length of a sentry's line-of-sight (which is set manually and assumed to be the same for all the sentries). It represents the maximum distance between a robot and a sentry where it can still be observed. If the sentries' line-of-sight is longer than the environment's diagonal length (l), then $d_{\max}$ is set to $l+1$, where l is the maximum possible distance between any two agents (this is important in order to have the best affect of the distance factor risk on the final path). $V(c)$ for a single sentry varies between 0 ($d_i = d_{\max}$) and 1 ($d_i = 0$). $V(c)$ is set to 0 for all cells that are not observed by any sentry.

In practice, the visibility map is generated by considering each sentry, one at a time. Initially all cells in the visibility map are set to 0. For each sentry, only those cells that are within its field of view are visited, using the Bresenham line-drawing algorithm[25] as an efficient raster scan method. For each direction from the sentry's cell position, the visibility cost is calculated for each cell along that direction until an obstacle or border is hit. The distance to the sentry (d_i) at a cell is simply the value of a counter that is incremented for every passed cell in a certain direction. The calculated cost at each cell is added to the previously stored value that sums up all the costs calculated for previously considered sentries.

Having the visibility map created, a covert path can be generated as described earlier in Section 4.2. Choosing a path with the minimum cost will find the right balance between the risk factors if conflicts occur; for example, the choice between a path exposed to two sentries at a far distance and a path exposed to one sentry at a close distance. Another factor that may conflict with the first two factors is the *exposure time*. A path exposed to a sentry at close distance for a short time (i.e., few cell steps) might be preferred over a path that is exposed at a far distance but for a longer time.

Discovering an unpredicted sentry(s) while moving along the created path can be easily accommodated. The path can be updated by updating the visibility map and repeating the

https://doi.org/10.1017/S0263574706002931 Published online by Cambridge University Press

cost-propagation process to generate a new path, given the robot's current location. Only the cells that are in the view field of the discovered sentry are updated in the visibility map. The calculated visibility cost of that sentry at each affected cell is added to the previously stored cost. The opposite process is applied in case one or more of the initially known sentries are not found on their locations (e.g., a sentry may have moved or its location might have only been suspected). The difference here is subtracting, instead of adding, from an affected cell's previous cost. In Section 7, a covert behavior will be discussed that allows choosing a *cautious* path when sentries are expected in the environment at unknown locations.

5.2. *Experimental results*

A simulator was built to test the DP algorithm for the case of complete knowledge. The simulator's inputs are the environment map (resolution of 200×120), the sentries' locations, the starting position, and the destination. Figures 1–3 show a number of test cases. The start and goal locations are represented by "R" and "G", respectively. The sentries' locations are represented by small white circles. The obstacles are shown in white. All the agents are assumed to be in a cell size for simplicity.

The generated covert path is the continuous line connecting the robot to the goal. In all the experiments presented in this paper, the shortest path to the goal is generated for evaluation purposes. The path is represented by a dashed line. The covert path is generated by setting α in (2) to a high value (e.g., 1000) for maximum possible covertness, while it is set to 0 to generate the shortest path. Moreover, the visibility map for each case is illustrated on the top of the environment map to evaluate the covertness of the generated paths visually. The visibility cost at each cell is represented in gray that gets darker as the cost gets lower, and vice versa.

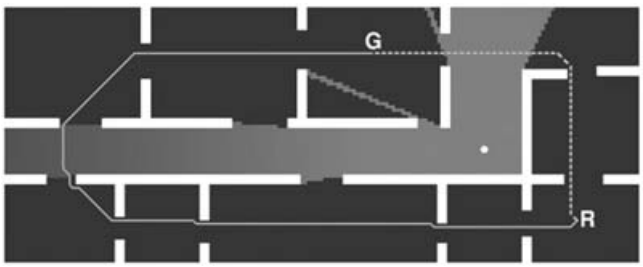


Fig. 1. The covert, continuous-line path between the robot R and the goal G has more covertness than the shortest, dashed-line path as it is exposed at far distance to the sentry located at the white dot. The obstacles are given in white.

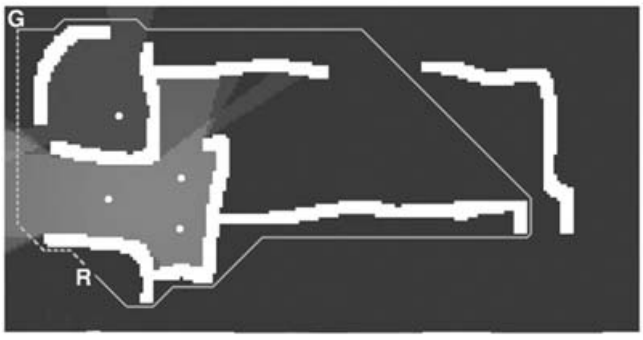


Fig. 2. The covert path is exposed to only one sentry, while the shortest path is exposed to all the sentries, which makes it a more risky path.

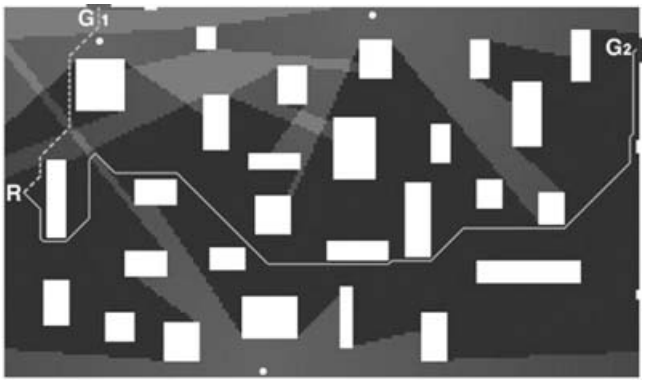


Fig. 3. The robot has to choose between two exit doors, G1 and G2. G2 is chosen as it offers a more covert path than the shorter path to G1.

Table I. Visibility risk measure of short and covert paths.

Figure	Short path	Covert path
1	18	5
2	19	2
3	15	0.6

It is noticeable in each example how the covert path is less exposed to the sentries than the shortest path (considering the described risk factors), where the covert path is mostly passes within a lower cost (darker) areas. Table I shows the covertness difference of the two generated paths. The measured visibility risk of each path is the summation of the calculated visibility costs for each cell along that path.

Figure 1 illustrates clearly the effect of the distance risk factor. The robot is exposed to the sentry at a far distance where it is more covert than the other shortcuts to the goal. Figure 2 illustrates clearly the risk factor associated with the sentries' number where the robot chooses a path that is exposed to only one sentry. The shortest path on the other hand is exposed to all sentries. A different case is shown in Fig. 3 for a multiple goals problem. The robot needs to reach a goal (e.g., an exit door in a hostile environment) that offers the most covert path (comparing the paths offered by other goals). The chosen path to goal G2 is more covert compared to the shortest path to G1, which is guarded by the sentries.

A straightforward extension to this case can provide another important covert behavior that allows a robot to covertly approach a hostile target. Let us assume that a robot has to approach and perhaps capture a target without giving it a chance to escape or take an offensive action, which may happen if the target detects the robot. One way to minimize such a risk is to approach the target taking as much care as possible not to be seen until reaching it, and thus to have the advantage of surprise. With no further modification to the approach, this situation is similar to the case of one known sentry (the target), where its location is the robot's goal.

https://doi.org/10.1017/S0263574706002931 Published online by Cambridge University Press

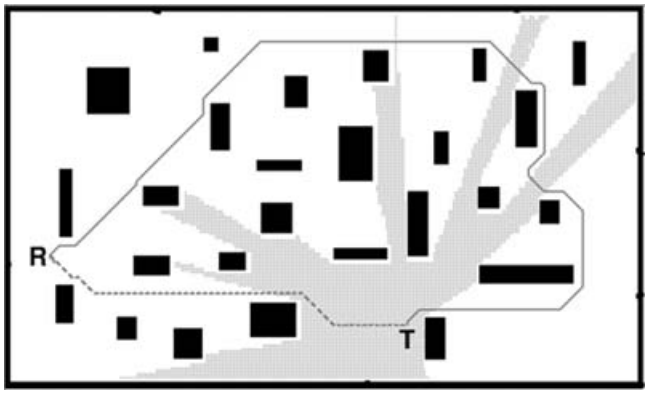


Fig. 4. The covert path to the target T is exposed to it at far, safe distance before it enters its field of view from where the path to reach it is the shortest. In contrast, the shortest path gives the target a better chance to escape.

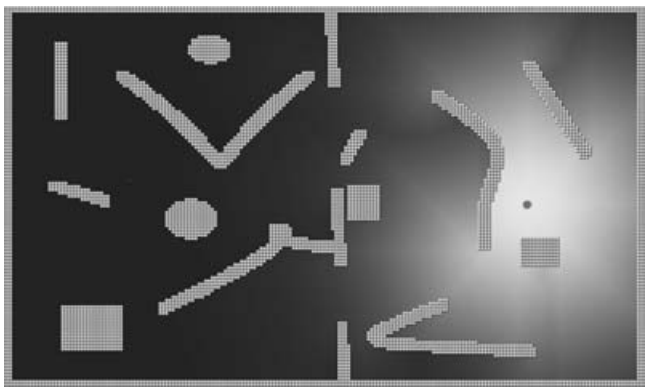


Fig. 5. A probabilistic model for a sentry located at the dark dot. The obstacles have been shaded. The brightness of each cell is proportional to its probability value. The level of uncertainty about the sentry's location is represented by the standard deviation value in the GD equation.

Figure 4 shows an example. Obstacles are in black and the target's view field is shaded. The robot has followed a stealthy path taking the risk of being exposed to the target at a far, but may be safe distance in order to enter the target's field of view from where the path to capture it is the shortest. On the short path, the robot has entered the target's view field from where it gives the target a better chance to escape. The measured visibility risk is 11 for the covert path, and 30 for the short one.

5.3. Uncertainty in the sentry's location

To cope with real-world uncertainties, the notion of probabilities is introduced to the DP algorithm to maximize the path covertness.[7] Sentries' initial given (or estimated) locations might be inaccurate or could possibly change if sentries can move. To take such considerations into account, the visibility map should be calculated incorporating those uncertainties.

The known Gaussian distribution (GD) is used to model the uncertainty of the sentries' locations. To date, the GD has only been applied in empty spaces. It has been modified[7] to accommodate the existence of the environment's obstacles. The probabilities are allowed to flow around the obstacles rather than penetrate them, thus, providing an accurate probabilistic model. An example is shown in Fig. 5. The

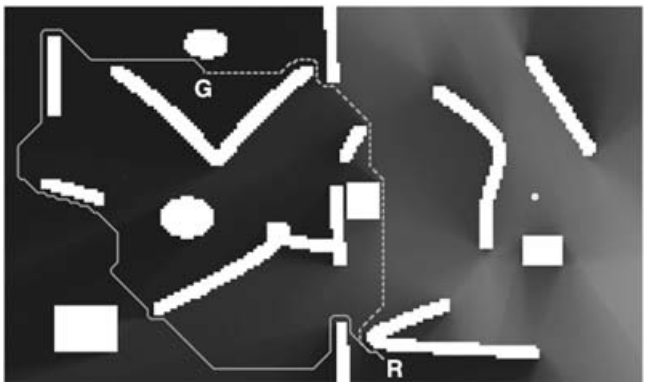


Fig. 6. The uncertainty of the sentry's location is represented in Fig. 5. The robot has chosen a path (the continuous line) that has less exposure to the high-probability area of the sentry's location.

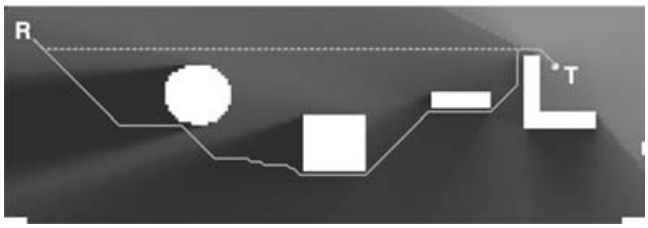


Fig. 7. The robot has followed a covert path (continuous line) to the target whose given location may be inaccurate or may change. The path minimizes the risk of being detected before getting close enough to capture the target.

created model is then used in the process of generating the visibility map. The details of the approach are presented in reference [7].

Two examples are shown in Figs. 6 and 7 for the case of avoiding detection by a sentry and the case of approaching a target covertly. The generated visibility map in each case is shown on the top of the free space. The short path in both the cases represents the covert path as well that would be generated if the given opponent's location is accurate. The generated covert paths are more cautious as the robot avoids exposure to the areas where the probability of the opponent is high.

6. Known sentries in unknown environment

6.1. Visibility map estimation

In an initially unknown (or partially known) static environment, a robot needs to find a path to the goal remaining unobserved for as long as possible, from the sentries whose known locations are assumed fixed. As in the previous section, the robot's knowledge about the sentries' locations might be available to it prior to navigation (e.g., suspected, or detected by a special device) or discovered later during the path. The work in this section can be compared to the work elsewhere,[22] which is described earlier in Section 2.2.

Generating a visibility map in this case will depend on the same factors described earlier. However, the calculated visibility cost at some cells will be an estimation rather than an accurate value, as a complete map knowledge is still unavailable. A visibility cost estimation method[6] is

https://doi.org/10.1017/S0263574706002931 Published online by Cambridge University Press

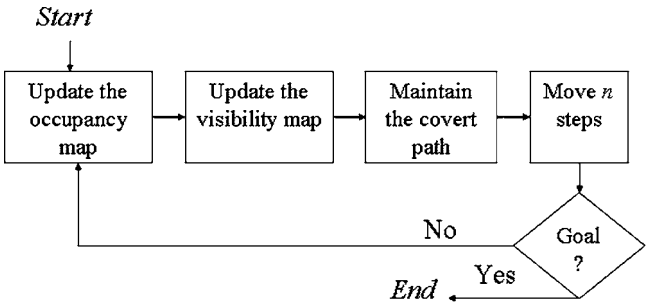


Fig. 8. The process of covert path planning in unknown environments. Four subtasks are repeated until the robot reaches the goal.

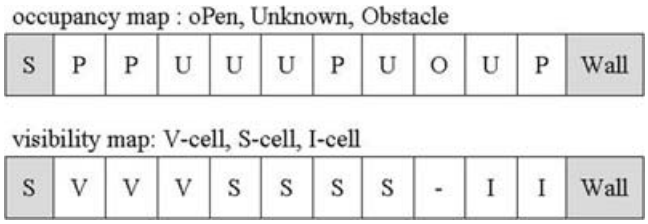


Fig. 9. A cell type in the visibility map depends on the occupancy states of its visibility line.

presented. It has provided promising results for this case. Both the visibility map and the occupancy map should be continuously updated during navigation to maintain the covertness and shortness of the robot's path. Hence, the robot behavior would be to balance between discovering a short path to the goal and simultaneously minimize its exposure to the sentries.

The path-planning method for this case is divided into four main subtasks, which are executed repeatedly until the robot reaches its destination (see Fig. 8). Initially, all the cells with an unknown state are marked as "unknown", which can be updated to either of the two states: "open" (i.e., free space) or "obstacle". Private visibility maps (*private maps*) are created for each sentry. These maps are updated one at a time when the general visibility map needs to be updated. The final visibility cost at each cell is the sum of the calculated costs in the related cell position in each of the private maps. For an unpredicted sentry, its private map is only created once it is discovered. Even the visibility costs calculated for each sentry can be added directly to the general visibility map (without the need for the private ones); however, this technique would be useful to reduce the computation burden of continuous updating, as will be described later.

In a private map, one of three *visibility types* is assigned to each cell that determine its visibility cost. The visibility type of a cell depends on the occupancy states on its visibility line (the visibility line is the set of all cells located on the imaginary line connecting the cell and the sentry). The first type is the invisible cells "I-cell", where a cell is invisible if its visibility line includes at least one obstacle cell. The visible cells "V-cell" is the second type, where a cell is known to be visible if the occupancy states at its visibility line are all open. The third type is the unsure cells "S-cell", categorized so if its visibility line includes no known obstacle cells and at least one cell with an unknown occupancy state.

In practice, determining the visibility type and calculating the visibility cost of each cell is done by moving along each direction from the sentry location and toward the environment border. Figure 9 shows an example of all cells located along one direction from the sentry (S) to a border cell (wall). The first row represents the occupancy states of the cells, while the second row represents the visibility types that are determined based on the first row. In this example, the cell next to the sentry is an open space; therefore, it is a V-cell. The next open cell is also a V-cell, as it is visible too. The next unknown cell is designated a V-cell, whether it is open or an obstacle cell. As we just passed an unknown cell, the following cells become of S-cells type until an obstacle cell is reached. The following occluded cells are marked as I-cells regardless of their occupancy states.

The visibility costs for each type are calculated as follows. I-cells get a visibility cost of zero. The visibility costs of V-cells are calculated using (4) as the cost would be similar as in the case of known environments. The costs at the S-cells are calculated with the same equation but with a modification that is presented in (5). The difference is the multiplication by the probability of that cell being a V-cell, which is inversely proportional to the number of unknown cells to the sentry. This probability can be calculated as q^m, where q is the probability of an unknown cell to be an open cell, while m is the number of unknown cells encountered on the visibility line. q is environment-dependent. One way to estimate q is to depend on the percentage of the open to the obstacle cells in the environment (in the later experiments, q is set to 0.5 assuming the probability of a cell being open or obstacle is the same).

$$V_u(c) = \left(1 - \frac{d_i}{d_{\max}}\right) \times q^m. \tag{5}$$

Using this equation, the calculated cost decreases significantly toward 0 as m increases. This estimation is realistic because as the number of unknown cells increases, the chance for at least one of them to be an obstacle (and hence the cell to be an I-cell) would increase too.

Having the visibility map generated by summing up all the private ones, the third subtask is to create the covert path given the updated visibility and occupancy maps. Using (2), the costs are propagated from the goal through both the open and the unknown cells. A path is created accordingly. The robot moves in that path for at least one step before next updating.

6.2. *The computation time*

As the number of sentries and the resolution of the environment increase, the burden of continuously updating the visibility map and the global path after each robot step may lead to unacceptable time complexity if the approach is executed as described earlier. A number of considerations are mentioned here that can significantly reduce the computation time to the point where the approach becomes acceptable for practical use.

It should be understood that both V-cells and I-cells' values in a sentry's private map do not need to be updated once they are assigned. Only the estimated values of S-cells are updated

https://doi.org/10.1017/S0263574706002931 Published online by Cambridge University Press

given the accumulated occupancy information. Moreover, updating a private map after each of the robot's step might not be necessary, where new gathered information at the step might not result in any changes. A private map updating is required only when at least one unknown cell (S-cell or V-cell type, i.e., its previous cost >0) is discovered to be an obstacle or an open space. Only the affected cells are recalculated in this case. If it is an obstacle, then all the cells it is occluding from the sentry (until the environment border or another obstacle is hit) will be designated as I-cells. If it is open, then all the cells following it should be updated using (5), until an obstacle or the border is hit. If more than one unknown cell (S-cell or V-cell) is discovered to be in the open state, then cells closer to the sentry are started with first to avoid updating some cells more than once.

Moreover, the visibility map does not need to be updated if the previously generated path includes only I-cells, where the path covertness is at its maximum level. Concurrently, the covert path updating is skipped when the visibility map is not updated, as the same path would be generated. An exception to that is only if an obstacle is detected on the current path. Nevertheless, the process of propagating the costs from the goal to all the connected free cells can be time consuming with high-resolution maps, an efficient cost-propagation method is presented later in Section 10.

To minimize the frequency of updating the visibility map and the covert path further, the robot can be allowed to move more than one step in its current path. For example, the robot can be allowed to continue moving on the same path while the next cell's visibility cost is less than a predetermined threshold, for a specified number of cell-steps.

6.3. Experimental results

Figures 10–12 show a number of test cases for the case of known sentries in an initially unknown environment. Both covert (continuous-line) and short (dashed-line) paths have been generated by setting α in (2) to a high value and to 0, respectively. The accurate visibility costs of the visibility map (which is initially unknown to the robot) are presented in grayscale. In all figures, it is noticeable how the discovered covert paths are less exposed to the sentries than the discovered short ones. A comparison between the two paths in terms of total visibility cost is presented in Table II (which is calculated similar to Table I).

In Fig. 10(a), the robot is shown to have successfully discovered a route that is completely invisible to all the sentries. Two earlier navigation phases are shown in Fig. 10(b) and (c), which highlight the decisions the robot makes with the information it has at hand. In Fig. 11, the robot has followed a path to a point (marked as "A") where it has changed its path to a more covert route, based on the

Table II. Visibility risk measure of short and covert paths.

Figure	Short path	Covert path
10	38	0
11	88	0.5
12	76	5

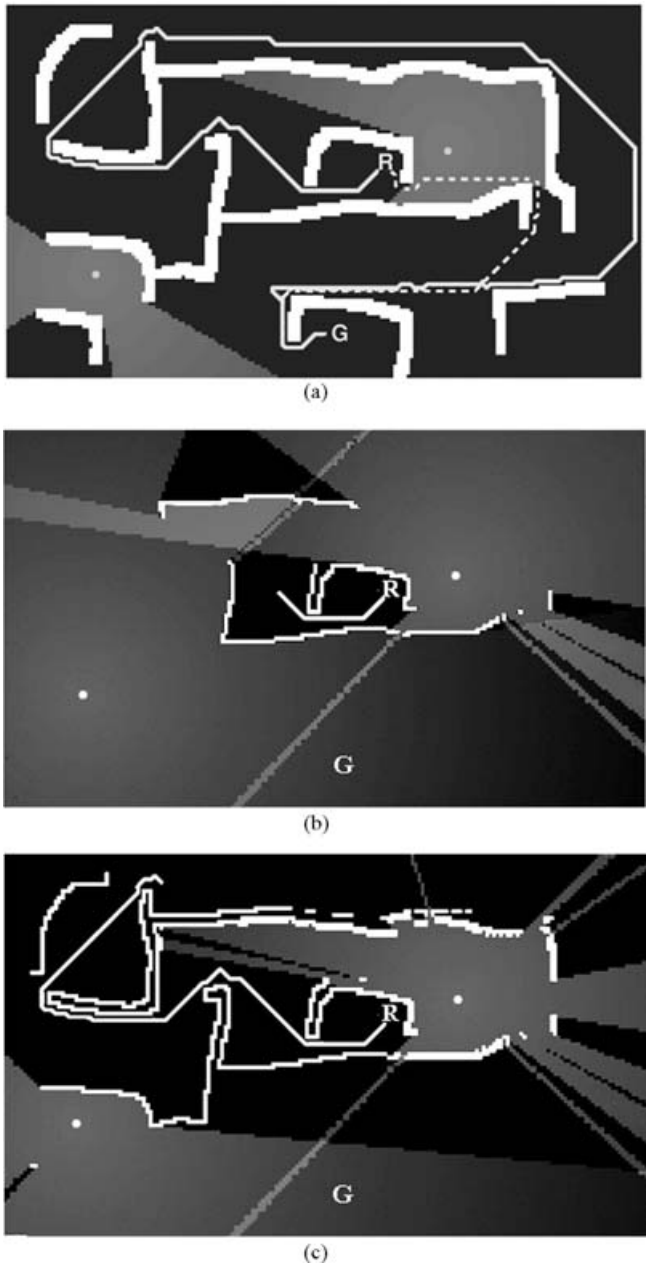


Fig. 10. (a) The robot has discovered a completely covert (continuous-line) path to the goal. A short (dashed-line) path is observed by one of the sentries. (b) and (c) Two earlier navigation phases.

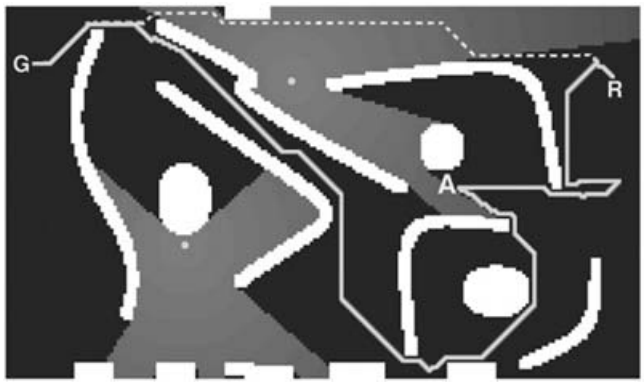


Fig. 11. At point "A" in the covert path, the robot has changed its course to follow another route that has more covertness.

https://doi.org/10.1017/S0263574706002931 Published online by Cambridge University Press



Fig. 12. The discovered short path is exposed to both the sentries, while the covert path is only exposed to one sentry at a far distance.

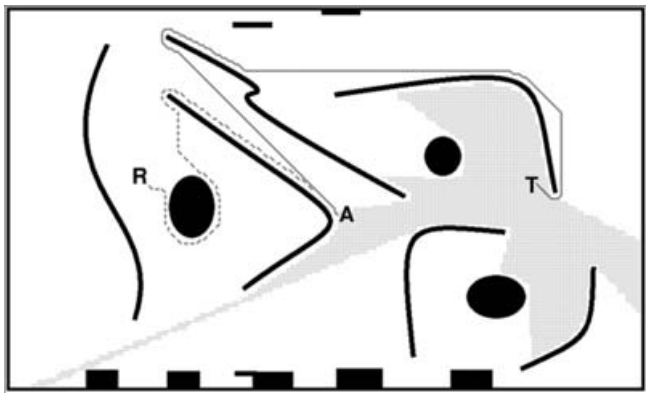


Fig. 13. The environment and the target's location are initially unknown. The robot has to search for it first (dashed line). A covert path (continuous line) to the target is created once it has been detected from point "A".

accumulated occupancy information. In Fig. 12, a completely covert path is not available. The robot has discovered and followed a reasonable covert path that is exposed to a sentry at a far distance.

Similar to the case in Fig. 4, the case of approaching a target covertly but in an initially unknown environment is presented in Fig. 13. To add more reality to the case, the target's location is assumed to be unknown. The robot has to search first for the target (using an off-shelf search method) where it has detected it from point "A" (the search path is the dashed line). A covert path to the target is planned accordingly. Yet, this approach can be extended to solve more complicated problems where the target is continuously moving, or where there are multiple robots or targets.

7. Unknown sentries in known environment

7.1. Visibility map estimation

To this point, we have discussed the situation when sentries' locations are initially known. The case of unpredicted sentries has been addressed as well, where they are taken into account only when detected. There might be a situation in which the robot may know or suspect that one or more sentries are around but does not know their positions. In this case, finding a cautious path to the goal that can minimize the risk of being observed would have less risk than just navigating a short path in which covertness is updated every time a sentry is detected. Moreover, the importance of a cautious path would be more obvious if some anticipated sentries are not detected on the path (e.g., the environment is monitored by hidden cameras), or they are actually moving around with no fixed positions.

Our solution to this problem[1] depends on the human covert behavior in such situations. A usual good strategy that can be noticed in real life is to follow a path with minimum exposure to free-space areas. The logical reason behind this strategy is that the probability of a position being exposed to a sentry at an unknown location would be proportional to its exposure to the free-space areas. A position that is exposed to the whole environment (e.g., the center of a star-shaped environment) is guaranteed to be exposed to any sentry. To implement this covert behavior with the DP algorithm, a different visibility map estimation method is used. The visibility map is found by calculating the exposure value of each cell as a risk factor. Each cell's risk cost will be in the range from 0 to the total number of free-space cells. Figures 14 and 15 in the next subsection show two different examples of a calculated visibility map where a cell gets darker as it becomes more hidden, and vice versa.

The basic way to calculate the exact exposure value at each cell is to use a ray shooting method. At any cell, the

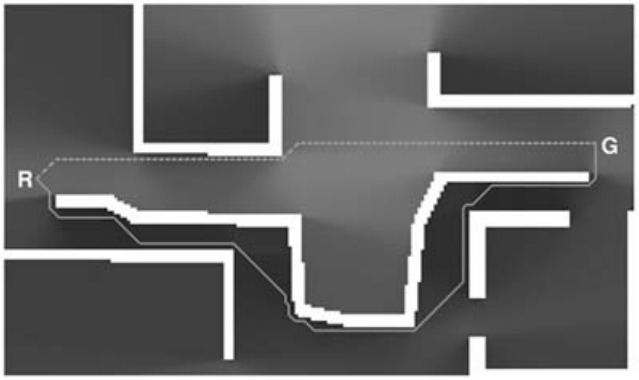


Fig. 14. The covert path is mostly passing through the low exposed areas. It is assumed to have a lower risk of being exposed to unknown sentries than the shortest path.

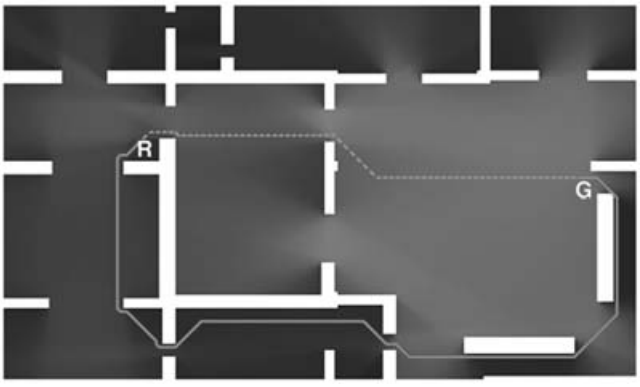


Fig. 15. The chosen covert path is relatively longer, but has a lower risk of exposure to free spaces where (possibly undetectable) sentries could be present.

https://doi.org/10.1017/S0263574706002931 Published online by Cambridge University Press

free-space cells are counted in each direction from that cell, where the exposure value is the sum of the counted cells in all directions. However, this process is time consuming for high-resolution maps as its time complexity is $O(n^2)$ in the worst case, where n is the number of free-space cells in the environment map.

A faster method is used based on the mutual visibility characteristic between cells. A counter is established at each cell that is incremented by 1 not only for every visited cell on each of its looking directions, but also when it is visited by a cell. Starting from the top-right cell in the map and ending with the left-bottom cell, only the first cell will need to count in all the directions. Each next cell, that has its turn to calculate its visibility, does not need to see in the direction of those cells whose visibility costs have already been calculated, clearly because they have already been counted. The number of directions decreases for each next cell where the exposure of the left-bottom cell will be already calculated by the time its turn comes.

The time consumed in generating both the visibility map and the covert path in the case of known environments is acceptable, as all calculations will be performed only once before the start of navigation (i.e., offline process). Moreover, the visibility map for an environment does not need to be recalculated if a covert path is required to be found between different start and goal positions.

A variation to the described case is if the probability of the sentries to be at any location, is not equal. The robot may know that a sentry has a higher probability to be in a place (e.g., a specific room) than in the other, or it may have a Gaussian distribution for a known, but uncertain, sentry location. Such knowledge can be implemented in the visibility map in the same way. Given a map in which each of the cells has a probability value of a sentry being there, the visibility cost at any cell will be the sum of the probability values at the cells exposed to it. This can be calculated using the above-described method where a cell's counter is incremented by the probability value of the visiting or visited cell.

7.2. Experimental results

Our strategy for the case of unknown sentries' locations is tested. Comparing the generated covert and short paths in Figs. 14 and 15, the covert paths have a lower risk of being observed as they are mostly passing through low-visibility (i.e., dark) areas. One feature that can be noticed in these paths is that they pass near the walls wherever possible. This behavior provides more covertness in reality. It also allows cautious self-exposing at corners to perhaps risky areas. In Table III, the percentage of the robot's average exposure to the entire environment along each path is calculated, which can show the advantage of the covert path over the short one.

Table III. Percentage of the robot's average exposure to the entire environment on its short and covert paths.

	Average exposure (%)	
Figure	Short path	Covert path
14	21	6
15	32	12
16	32	20

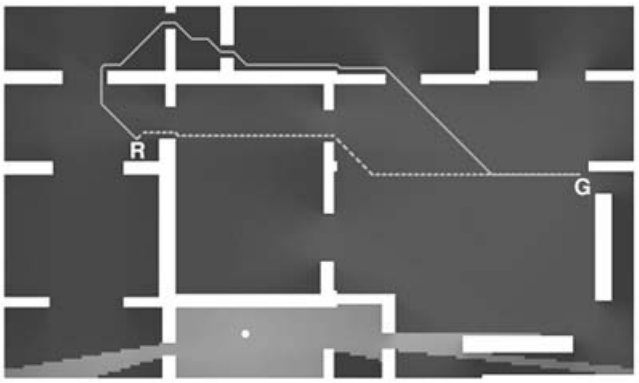


Fig. 16. The visibility cost of cells observed by the known sentry is the sum of their general and special visibility costs. The covert path is hidden from the sentry but still has its general covertness. (*Note:* The changes in the cost level may not be noticeable in this figure as the color range is limited.).

Figure 16 shows a case similar to Fig. 15 in terms of environment map and start/goal locations. The difference is that one of the sentries' locations is initially known. In this case, the visibility cost of a cell observed by the known sentry will be the sum of its exposure value (general covertness) and the sentry's visibility cost (special covertness) that has been given a higher weight for its obvious risk. The robot has chosen a different path than the one in Fig. 15. The new path is hidden from the known sentry but is still cautious from the unknown ones, as compared to the short path.

8. Unknown sentries in unknown environment

8.1. Visibility map estimation

The highest level of the complexity hierarchy for covert path planning is to start navigation with absolutely no prior knowledge about both the sentries and the environment. The robot needs to discover a path to the goal while simultaneously minimizing its exposure to the open and unknown areas, where expected sentries might be found (in Section 6, no sentries are expected in the environment other than the known ones). The strategy used here is similar to the one in Section 7, which is to minimize exposure to free spaces. The difference here is to start with the assumption that unknown cells are free spaces unless discovered to be otherwise. This assumption is not only essential to find a path to the goal, but also provides a worst-case scenario in terms of the sentries' possible locations. The visibility cost of a cell will be its exposure to both free and unknown spaces.

Path planning in this case is an online process, where continuous updating of the visibility map and the covert path is required as the map knowledge is accumulated (Fig. 8, Section 6). Initially, the visibility costs at all the cells are set to the maximum cost (map's width $w \times$ height h), as all the cells in the occupancy map are initially unknown. As the map is updated with the locations of the obstacles, the visibility

https://doi.org/10.1017/S0263574706002931 Published online by Cambridge University Press

costs (i.e., exposure values) of the affected cells decrease accordingly.

The time complexity associated with the visibility map updation process using the method described in Section 7 is unacceptable for the case of unknown environments. In reference [5], a faster method is presented that depends on triangulation geometry to find an approximate exposure value of each cell. The inaccuracies involved in such an approximation do dissolve during the cost-propagation process when applying Eq. (2). The time complexity for one update of the visibility map is $O(f \times a)$, where f is the number of affected cells, and a is the required level of the exposure accuracy. The details of this method are presented elsewhere.[5]

8.2. Experimental results

The test cases in Figs. 17 and 18 represent the case of no initial knowledge about the environment and the sentries' locations (sentries are assumed undetectable). The displayed visibility maps in these figures are accurate, which are initially unknown to the robot. The robot's navigational behavior can be described as taking the advantage of the discovered low-visibility areas to reduce its exposure on its way to the goal, which maximizes its covertness. The average exposure of the covert and short paths to the entire environment is calculated for each example and presented in Table IV.

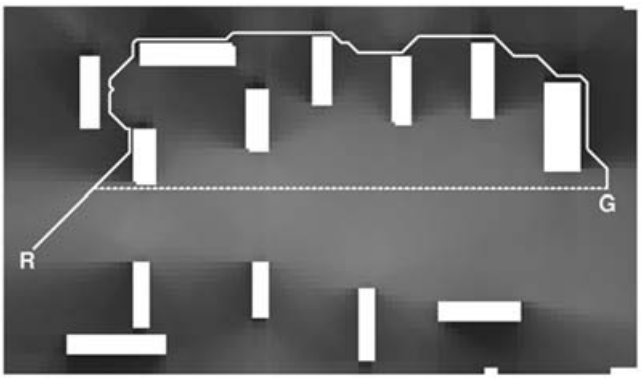


Fig. 17. The robot has discovered a path with low exposure compared to the discovered short path.

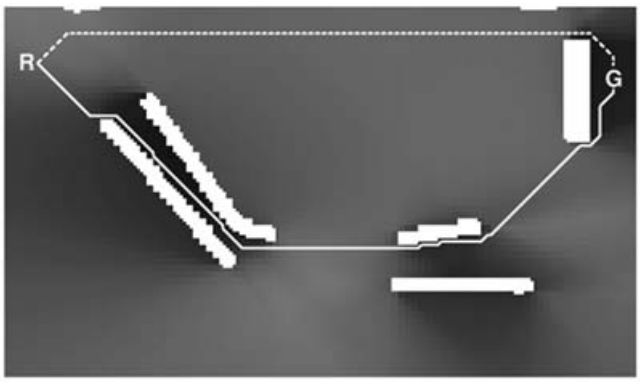


Fig. 18. The robot has passed through two tunnels to minimize its exposure to the free spaces.

Table IV. Percentage of the robot's average exposure to the entire environment on its short and covert paths.

	Average exposure (%)	
Figure	Short path	Covert path
17	44	19
18	50	33

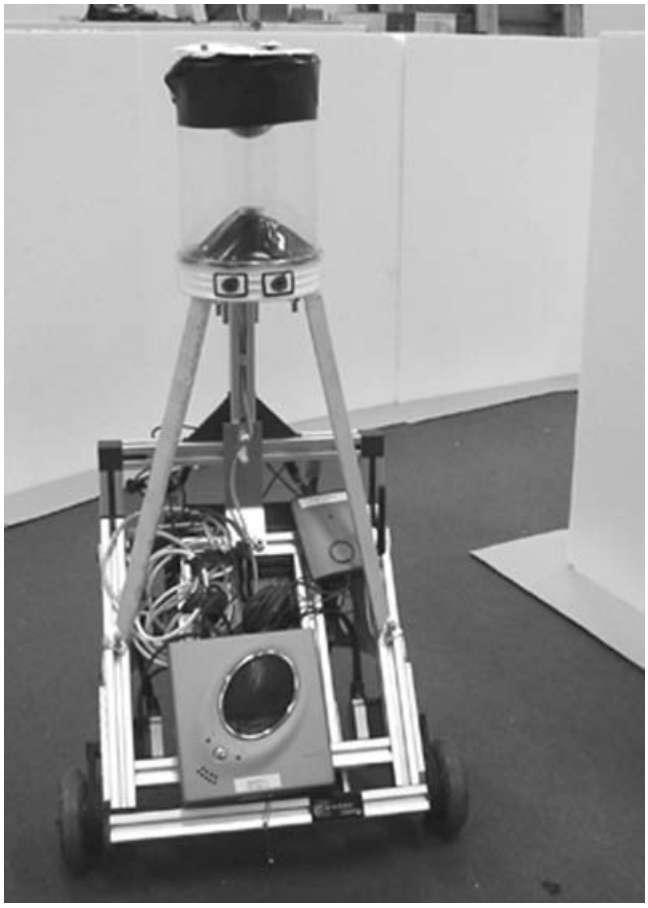


Fig. 19. Panoramic vision system is added to the ER1 robot to acquire a 360° view of the environment.

In more cluttered environments, an additional behavior that would be required is to look for better chances of covertness by discovering, instead of just taking the chance at hand. However, an added discovery behavior would involve the risk of being unnecessarily exposed to some areas. Further research should be undertaken to identify when it is less risky to discover more, or when it is good to just continue navigating in the current covert path.

9. Physical experiments

The DP algorithm has been tested with a physical experiment to confirm its applicability in real-world environments. The robot used is a modified version of ER1 robot, which is built by the Evolution Robotics Company. The robot is equipped with panoramic vision to acquire a 360° view of the environment (see Fig. 19). The vision system consists of a specially shaped curved mirror and a conventional camera.[29]

https://doi.org/10.1017/S0263574706002931 Published online by Cambridge University Press

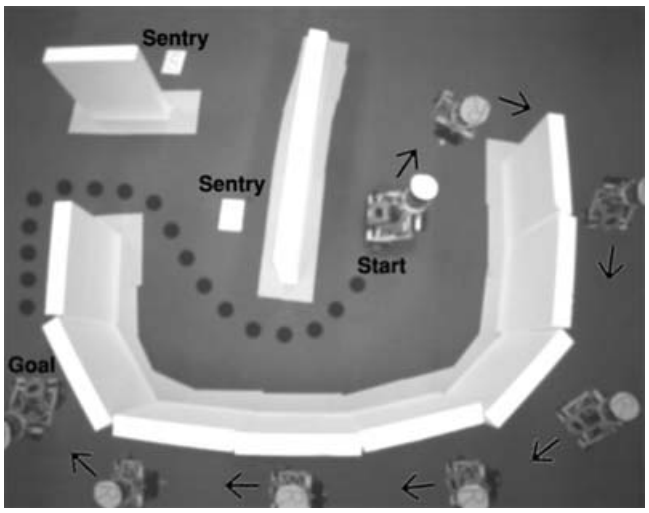


Fig. 20. Top view of a physical experiment. A covert path from "Start" to "Goal" should be discovered, given two known "Sentries". The robot's navigation phases show that it has found a completely covert path. The shortest (dotted-line) path is mostly visible to one of the sentries.

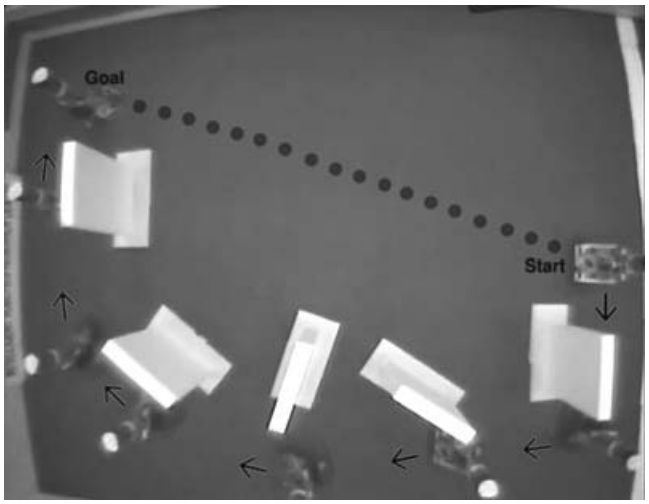


Fig. 21. This experiment shows the case of unknown, undetectable sentries. The robot makes use of the scattered obstacles to minimize its exposure to the free spaces. A short path on the other hand has a higher risk of exposure.

The experiment was conducted in a robot soccer field in an indoor environment (5 m × 6 m). The obstacles used were opaque and higher than the robot's sight level. The robot depended on its odometry for localization, which was sufficient for the given environment size. For the purpose of mapping and sentry detection, the robot depended on the color differences of the free spaces (green), obstacles (pink), and sentries (white).

Two experiments are presented in Figs. 20 and 21, where the environment is initially unknown in each case. Figure 20 shows the case of known sentries (Section 6). Obstacles are shown in white with a pink base. The robot (initially located at "Start") needs to discover a path to the goal ("Goal") but with minimum exposure to two known sentries located at "Sentry". The robot has discovered in real time a completely covert path, which is represented by the robot's different navigation phases on that path. A comparative short path (the dotted line) is visible to one of the sentries as can be judged visually. In Fig. 21, the sentries' locations are unknown. The robot uses the seen obstacles to navigate covertly to the goal. In contrast, the short path has a higher risk of being visible to the possibly undetectable sentries.

In the previous simulation experiments, it is assumed that the time the robot takes to move from one cell to another is fixed. In contrast, a physical robot may not be able to do that. For example, the ER1 robot would stop at some cells to rotate and change its motion direction; this leads to the spending of a longer time at some cells. Hence, the actual cost of the traveled path might be higher than what has been calculated. A further investigation to understand how a robot's dynamics can be integrated in the system would be important, since the approach is meant for physical robots.

10. The DP algorithm computation time

10.1. Fast visibility–distance cost transformation method

The concern about the time complexity of the DP algorithm lies in the case of unknown environments where there is a need to frequently update both the visibility map and the global covert path. The computation time is proportional to the map resolution, but is inversely proportional to the robot's knowledge of the environment, where the rate of updation decreases as knowledge gets accumulated.

Fast visibility map updating methods has been mentioned earlier in Sections 6 and 8, which depend on only updating the affected cells instead of the entire map. Here, the second part of the concern on the time complexity is discussed. The process of updating the covert path can be accelerated. As described in Section 4, the DP algorithm finds the visibility–distance cost at all nonobstacle cells. A better approach is to use the known Dijkstra's algorithm (uniform cost search) to expand the search for a path only through those cells that are highly expected to be part of the final path. Starting from the goal location, the cell with the lowest calculated visibility–distance cost to the goal is expanded first. The search process can be accelerated further by expanding the costs simultaneously from the robot side (bidirectional search).

An example is shown in Fig. 22 for the case of unknown sentries in known environment (the visibility cost at any cell is its exposure). The dotted areas represent the only cells that got a visibility–distance cost, which are 29% of the total free space. The cost has propagated from the robot and the goal locations mostly through the low-visibility route.

10.2. Recorded computation time

An approximated computation time of the DP algorithm in unknown environments is measured. The test cases presented in the previous sections have been simulated on a laptop with a processing speed of 1.5 GHz Mobile Pentium. The environment's map resolution is 200 × 120. The average robot's speed in the case of two known sentries' locations (Figs. 10–12 in Section 6) is 95 stepped cells per second. The average robot's speed in the case of unknown sentries (Figs. 17 and 18 in Section 8) is 136 cells per second. In these measurements, the processing speed is only affected

https://doi.org/10.1017/S0263574706002931 Published online by Cambridge University Press

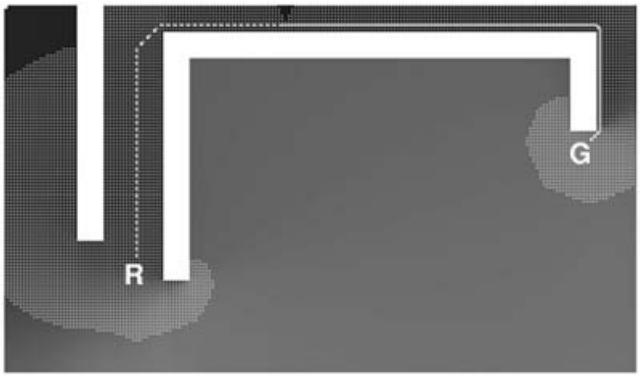


Fig. 22. The visibility–distance cost is propagated from both the ends, R and G. The dotted areas represent the expanded cells. At the collision point, the (dashed-line) path to the robot and the (continuous-line) path to the goal are connected forming the final path.

by the visibility map and the covert path updation (the path is updated after each cell-step, if required). In the conducted physical experiments (Section 9), the motion speed of the robot was limited by its motion constraints (e.g., its maximum speed), not by the computation time of its planner.

Even the concern about time complexity in 2-D path-planning problems has declined in this century due to the rapid increase in hardware processing speed. However, more research would be required to improve the algorithm's time complexity for very large-scale environments.

11. The DP algorithm for other applications

The DP algorithm can be a useful tool for solving other visibility-based path-planning problems by introducing the appropriate visibility cost factors. The exact negative picture of the presented approach is a solution for *overt path-planning* problems. In this case, the robot seeks visibility instead of avoiding it by planning a path with maximum exposure (to all free spaces or to specified locations). Applications that involve searching or exploring can benefit from this approach. For example, a robot that is searching for a target might have a higher chance of finding it if it navigates to the most visible point through the most visible path. In references [7, 30], the DP algorithm is presented as a solution for a target-tracking problem. A mobile robot should keep an unpredictably moving target in a cluttered environment within its field of view.

A different kind of application that has benefited from the DP algorithm is presented elsewhere.[31] The robot needs to find a path in high-risk fire-front environment with minimum exposure to flames and heat. A safe path is generated that takes advantage of the surrounding occluding obstacles as a heat screen. Both escape routes and fire-extinguishing forays can be planned using this methodology.

12. Conclusion and future work

In this paper, a new visibility-based path-planning approach was introduced (the DP algorithm). It has been presented as a solution for covert navigation problems. A number of simulated and physical experiments are illustrated to overcome the problem of covert path planning to a specified destination. The approach has shown its capability of adapting with different levels of knowledge about both the operating environment and the sentries' locations. For any level of initial knowledge, a path with considerable covertness can be generated compared to a short path to the same destination. Moreover, the approach has been presented as a solution for different kind of applications.

This paper leaves numerous important extensions open for further research. More realistic assumptions about the operating environment or the robot's/sentries' capabilities should be investigated. Cooperative covert robot is an interesting extension that should be investigate. Different navigation and communication strategies should be developed to achieve different team-based covert missions. Moreover, the approach should be extended to work in dynamic environments.

Some security organizations might have conducted private researches on covert robotics. From a researcher's perspective, such work cannot be recognized until it is published. Given the existing work, the covert robotics field is yet to grow. Various abilities should be created and integrated to deal with complicated tasks at the required level of covertness.

Acknowledgments

The authors would like to thank Etisalat University College and the Emirates Telecommunication Corporation (Etisalat), United Arab Emirates, as well as the Intelligent Robotics Research Centre (IRRC) as part of the Australian Research Council Centre for Perceptive and Intelligent Machines in Complex Environments, Monash University, for providing the financial support for this work.

References

1. M. Marzouqi and R. A. Jarvis, "Covert path planning for autonomous robot navigation in known environments," *Proceedings of the Australasian Conference on Robotics and Automation (ACRA)*, Brisbane, Australia (Dec. 2003).
2. M. Marzouqi and R. A. Jarvis, "Covert robotics: Hiding in known environments," *Proceedings of the IEEE Conference on Robotics, Automation and Mechatronics (RAM)*, Singapore (Dec. 2004).
3. M. Marzouqi and R. A. Jarvis, "Covert robotics: Covert path planning in unknown environments," *Proceedings of the Australasian Conference on Robotics and Automation (ACRA)*, Canberra, Australia (Dec. 2004).
4. R. A. Jarvis, "Distance transform based visibility measures for covert path planning in known but dynamic environments," *Proceedings of the 2nd International Conference on Autonomous Robots and Agents (ICARA)*, New Zealand (Dec. 2004).
5. M. Marzouqi and R. A. Jarvis, "Fast visibility evaluation for covert robotics path planning," *Proceedings of the IEEE International Workshop on Safety, Security and Rescue Robotics (SSRR)*, Japan (Jun. 2005).
6. M. Marzouqi and R. A. Jarvis, "Covert path planning in unknown environment with known or suspected sentry location," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Canada (Aug. 2005).

https://doi.org/10.1017/S0263574706002931 Published online by Cambridge University Press

7. M. Marzouqi and R. A. Jarvis, "Accommodating uncertainty in covert and overt robot path planning," *Proceedings of the IEEE TENCON (IEEE International Region 10 Conference)*, Melbourne, Australia (Nov. 2005).
8. J. F. Canny and J. H. Reif, "New lower bound techniques for robot motion planning problems," *Proceedings of the 28th IEEE Symposium on Foundations of Computer Science*, Los Angeles, CA (1987) pp. 49–60.
9. O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *Int. J. Robot. Res.* **5**(1), 90–98 (1986).
10. S. M. LaValle, Rapidly-Exploring Random Trees: A New Tool for Path Planning. *Technical Report 98–11* (Ames, IA: Computer Science Dept., Iowa State University (Oct. 1998).
11. B. Kratochvil, Implementation of Scout Behaviors Using Analog Sensing Methods. *Technical Report 03–044* (Morris: University of Minnesota (2003).
12. P. E. Rybski et al., "A team of robotic agents for surveillance," *Proceedings of the 4th International Conference on Autonomous Agents (ACM)*, Barcelona, Spain (2000) pp. 9–16.
13. J. M. McMichael and M. S. Francis, *Micro Air Vehicles Toward A New Dimension in Flight* (*Defense Advanced Research Projects Agency (DARPA), USA* (1997).
14. R. T. Vaughan, G. S. Sukhatme, F. J. Mesa-Martinez and J. F. Montgomery, "Fly spy: Lightweight localization and target tracking for cooperating air and ground robots," *Distrib. Auton. Robot. Syst.* **4**, 315–324 (2000).
15. S. M. LaValle, H. H. Gonz'alez-Ba, C. Becker and J.-C. Latombe, "Motion strategies for maintaining visibility of a moving target," *Proceedings of the IEEE International Conference on Robotics and Automation*, Albuquerque, NM (1997) pp. 731–736.
16. K. Mann, "Evolving robot behavior to play hide and seek," *J. Comput. Small Colleges Arch.* **18**, 257–258 (2003).
17. B. M. Moore, C. Geib, and B. Reich, "Planning for reactive behaviors in hide and seek," *Proceedings of the 5th Conference on Computer Generated Forces and Behavioral Representation*, Orlando, FL (May 1995).
18. I. Suzuki and M. Yamashita, "Searching for a mobile intruder in a polygonal region," *SIAM J. Comput.* **21**(5), 863–888 (1992).
19. R. A. Jarvis, "Collision-free path trajectory using distance transforms," *Proceedings of the National Conference and Exhibition on Robotics,* Melbourne, Australia (Aug. 1984) and *Mech. Eng. Trans. J. Inst. Eng.* (1985).
20. S. Ravela, R. Weiss, B. Draper, B. Pinette, A. Hanson and E. Riseman, "Stealth navigation: Planning and behaviors," *Proceedings of the ARPA Image Understanding Workshop*, Monterey, CA (1994) pp. 1093–1100.
21. Y. Teng, D. DeMenthon and L. Davis, "Stealth terrain navigation," *IEEE Trans. Syst. Man Cybern.* **23**(1), 96–110 (1993).
22. E. Birgesson, A. Howard and G. Sukhatme, "Towards stealthy behaviors," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas (Oct. 2003) pp. 1703–1708.
23. A. Tews, M. J. Mataric and G. S. Sukhatme, "Avoiding detection in a dynamic environment," *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, Sendai, Japan, Sep. 28–Oct. 2 (2004).
24. J. Barraquand, B. Langlois and J. C. Latombe, Numerical Potential Field Techniques for Robot Path Planning. *Report No. STAN-CS-89–1285* (Stanford, CA: Department of Computer Science, Stanford University (1989).
25. J. Bresenham, "Algorithm for computer control of a digital plotter," *IBM Syst. J.* **4**(1), 25–30 (1965).
26. T. Bandyopadhyay, Y. Li, M. H. Ang Jr. and D. Hsu1, "Stealth tracking of an unpredictable target among obstacles," *Proceedings of the International Workshop on the Algorithmic Foundations of Robotics*, The Netherlands (2004).
27. A. Zelinsky, "Using path transform to guide the search for findpath in 2d," *Int. J. Robot. Res.* **13**(4), 315–325 (1994).
28. A. Rosenfeld and J. L. Pfaltz, "Sequential operations in digital image processing," *J. Assoc. Comput. Mach.* **13**(4), 471–494 (1966).
29. J. S. Chahl and M. V. Srinivasan, "Reflective surfaces for panoramic imaging," *Appl. Optics* **36**(31), 8275–8285 (1997).
30. M. Marzouqi and R. A. Jarvis, "Distance transform based Gaussian distribution for probabilistic target tracking," *Proceedings of the IEEE/RJS International Conference on Intelligent Robots and Systems (IROS)*, China (2006), in press.
31. R. Jarvis and M. Marzouqi, "Robot path planning in high risk fire front environments," *Proceedings of the IEEE TENCON (IEEE International Region 10 Conference)*, Melbourne, Australia (Nov. 2005).
32. A. Zelinsky, "A mobile robot exploration algorithm," *IEEE Trans. Robot. Autom.* **8**(6) (1992).
33. A. Stentz, "The focussed D* algorithm for real-time replanning," *Proceedings of the International Joint Conference on Artificial Intelligence* (Aug. 1995).

https://doi.org/10.1017/S0263574706002931 Published online by Cambridge University Press