To appear in Proceedings of the International Conference on Intelligent Robots and Systems, Sendai, Japan, Sep-Oct 2004

Avoiding Detection in a Dynamic Environment

Ashley Tews, Maja J. Matarić and Gaurav S. Sukhatme
Computer Science Department
University of Southern California
Los Angeles, USA
atews|maja|gaurav@robotics.usc.edu

Abstract— **Remaining elusive while navigating to a goal in a dynamic environment containing an observer requires taking advantage of opportunistic cover as it occurs. A reactive navigation approach is needed that recognizes the utility of environment features in offering protective cover. We present an approach that allows stealthy traverses in unknown environments containing dynamic objects. It is a frontier-based method that allows a robot to follow in the obscuring shadow of objects despite their dynamics, and take advantage of more opportunistic cover if it becomes available. An analysis of our approach in off-line modeling and experiments conducted in simulation and outdoor environments demonstrate its effectiveness in achieving high quality solutions for stealthy navigation.**

Keywords-component; stealthy navigation; dynamic environment; mobile robot

I. Introduction

Enabling robots to be stealthy during navigation reduces their risk of exposure to unwanted observers. The observers should remain unaware and unaffected by the robot's activities. There are many military and security applications for stealthy navigation in areas of reconnaissance, scouting and surveillance, or for safe autonomous transport of payloads or people in observable areas. More benign applications include allowing maintenance robots to remain relatively obscure from people, such as cleaning robots in public areas. In each of these applications, the robot needs to react to the possibly unknown environment and its objects. Static objects can exist as natural or artificial structures that allow the robot to take potentially discrete paths around them. Dynamic objects offering reasonable coverage may manifest in the environment as vehicles or other robots. The navigation algorithm should be able to evaluate and take advantage of each of these types of objects if they prove beneficial to the robot's task.

In our previous work [1], we demonstrated a method for stealthy multi-robot navigation in the presence of an observer using static objects in outdoor environments. In contrast, the goal of the research presented here is to determine how robots can take advantage of dynamic objects in similar types of environments.

This work is supported in part by DARPA grants DABT63-99-1-0015, and 5-39509-A under the Mobile Autonomous Robot Software (MARS) program.

Our approach to solving the general case of reactive stealthy navigation is to define a cost function that embodies the parameters of 'stealth' and 'efficiency'. Stealth is defined as the ability to maintain a low profile during navigation in the presence of an observer. Efficiency is defined in terms of minimizing the length of the navigation path to a goal. We encode the cost function as parameterized potential fields that model features of the environment and the task. This approach has demonstrated good results in our previous work, however it does not account for dynamic objects in the environment. The extension discussed in this paper allows the robot to capitalize on mobile objects offering significant cover during its traverse. The approach is model-free and makes few assumptions about the dynamics of the objects in the environment. It is demonstrated in simulation and in outdoor environments using a Pioneer AT and a Segway Robotic Mobility Platform (RMP). It is also evaluated against empirically-defined cost functions that evaluate completed paths for their efficiency and stealth performance. The results demonstrate our algorithm achieves high performance according to the criteria outlined in the cost function.

An analysis of related work is presented in Section II followed by a discussion of our approach in Section III. Sections IV and V outline the details of experiments carried out in simulation and outdoor environments, respectively. Section VI provides an evaluation of empirically-optimal stealthy and efficient paths for a given environment, and how our approach compares. Section VII presents a summary of the approach .

II. Related Work

There has been little research conducted into low-visibility path planning for mobile robots in outdoor environments. Mostly it involves the use of a priori maps and observers with known locations. [2] discretizes the environment into cells that are assigned to virtual processors. The processors compute the visibility constraints of each cell using information about the mobile observers in the environment. Combining this information allows a reactive path to be determined for the robot. [3] analyzes digitized terrain features for visual servoing to a goal in the presence of an observer. [4] models observers and potential navigation waypoints using virtual springs and masses. The system stabilizes to generate a low-visibility path for an unmanned air vehicle crossing an area containing multiple radar sites. [5] uses probabilistic cost functions for balancing information gain versus the probability of exposure in determining static observation positions for target surveillance. This concept is also used for distributing the sensor load between vehicles in formation to maximize observation during navigation.

Of the approaches for low-visibility path planning without the use of a priori maps, [6] presents a reactive method for a robot to use stationary objects in the environment for cover while navigating to a goal. [1] extends this concept for multiple robots conducting sequential traverses. Path quality is improved on successive traverses from the integration of environment and path information from the preceding robot.

In each case above, the research was conducted in either static environments or environments where the only dynamic objects were the robots themselves or the observers. In many cases, the researchers indicated that the most important criteria for developing low visibility traverses in the presence of observers are to determine the shortest possible path that offers the least exposure to the observers. Our work focuses on these criteria and allows the environment to contain dynamic objects beneficial for providing cover to the robot during its traverse.

III. THE APPROACH

In our previous work [1], task and environment-related information is combined to embody the stealth-efficiency cost function. This information consists of:

- objects in the environment modeled on an occupancy grid [7] developed during the robot's traverse,
- their effect on the robot's task by means of providing a 'shadow' from the observer's position for the robot to hide in, and
- task-related information in the environment including the distances to the observer, goal, and robot.

Each element is defined by a parameterized function whose value determines its effect on the robot's path planning decisions. The functions are represented as potential fields [8][9] which are modeled as either low-valued attractors (e.g., distance to the goal) or high-valued repellors (e.g., distance from the observer) and most have a global affect across the environment. By combining fields, a task-oriented view of the environment is generated and a global minimum is extracted as the next navigation waypoint. This minimum is the centroid of the lowest-valued region. A small region is chosen rather than a point to alleviate the canonical local minima and oscillation problems inherent in potential field methods, and to filter the effects of sensor noise altering the occupancy grid.

By using potential fields in this manner, the robot is able to integrate new information about the environment as it is sensed and use it for reactive decisions about the next waypoint to traverse to. Previous results [1] demonstrate that this approach produces intuitive low-visibility paths in unknown static environments.

The extension discussed in this paper allows the robot to reactively take advantage of opportunistic moving cover offered by objects in the environment. There are three components introduced to the existing algorithm to enable this capability: 1) a safety zone in an object's shadow, 2) a frontier timeout that allows the robot to remain stationary in this zone for a period of time before continuing, and 3) the ability to capitalize on better opportunities as they occur.

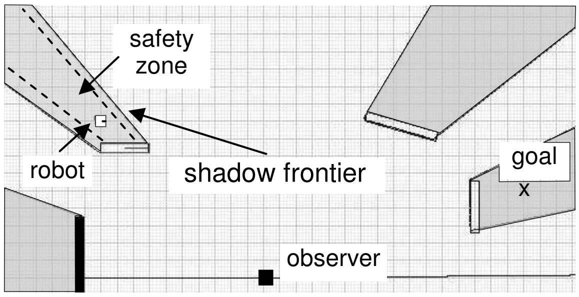

Figure 1. The shadow regions are cast behind objects from the observer's position. Safety zones are regions inside shadows for the robot to manoeuvre without being exposed by accidentally overrunning a frontier.

1) *Safety Zone*

Each object in the environment typically casts a coverage 'shadow' that represent an area where the robot is obscured from the observer. This area is bounded by frontiers that separate visible space from obscured. A *safety zone* (Figure 1) is defined as an area within a shadow where a robot can safely manoeuvre without fear of accidental detection. It is offset from the shadow frontier closest to the robot and is wide enough to reduce the effect of sensor noise and the possibility of the robot overrunning the shadow frontier while stopping or turning. A robot in a safety zone will effectively remain at a relative distance from the shadow frontier. Therefore, if the object moves, so does the zone and the robot dynamically maintains cover as long as the object travels within the velocity constraints of the robot and in a direction that satisfies the embedded stealth-efficiency cost function.

2) *Frontier Timeout*

The purpose of the frontier timeout is to allow the robot to remain stationary for a period of time to see if the object moves. This allows the robot to utilize an object that stops for limited periods of time. Upon a timeout, the robot will navigate to its next waypoint. The value of the timeout is empirically set for a reasonable waiting time aligned with the time-constraints of the traverse.

3) *Opportunistic Waypoints*

During the robot's traverse, it constantly interrogates the environment to find regions offering better coverage. This operation is also active when the robot is stationary in a safety zone. If the environment within the robot's sensor range changes to offer a more appealing path, the robot will forgo its current cover in lieu of the new opportunity. This occurs if the difference in potential value between the current waypoint and the one calculated with new environment information is beyond a threshold value. The threshold is empirically determined to account for small perturbations in sensor information.

The algorithm incorporating these components is shown below (Figure 2). By using frontier-following and opportunistic waypoints, the robot is highly responsive to changes in the environment that benefit its task. The constraints on the usefulness of dynamic objects are that they should:

- offer a potential benefit to the robot's objective, otherwise it will be ignored,
- move slower than the robot's maximum velocity, and
- stop for a shorter amount of time than the robot's frontier timeout, if they stop at all.

```
Update the occupancy grid with new sensor data
Generate and combine potential fields from the occupancy grid, observer, goal and robot's locations
Extract the lowest-valued region's centroid as the next waypoint
If assumed detected by the observer
  Navigate to the next waypoint
Else
  If in a safety zone and moving
    Stop near the shadow frontier
    Start the frontier timer countdown
    Save the current waypoint and its potential
      field value
  Else
    If ((current waypoint value - next waypoint value) >
    threshold) OR (frontier timer = 0) OR (not in the
    safety zone))
        Navigate to the next waypoint
    End
  End
End
```

Figure 2. The stealthy navigation algorithm.

In the following sections, we present results of experiments conducted in simulation and outdoor environments to validate the approach. The simulation results demonstrate the approach in its entirety (i.e., frontier-following, frontier timeouts and opportunistic waypoint selection) in an environment where the moving object has a varying velocity. The real world experiments are in similarly configured outdoor environments and demonstrate the frontier-following technique and timeouts on a Pioneer AT and Segway RMP with an object that moves at a constant velocity. In all experiments, the observer has infinite range, omni-directional sensing.

IV. SIMULATION EXPERIMENTS

The simulation experiments were conducted using Player devices and the Stage simulator [10]. The environment measures 35m by 35m and is configured as shown in Figure 3. It consists of three static and one mobile barricade with the linear reversible path shown in the figure. The mobile barricade starts where shown and reverses for 10m, changes direction and stops at random intervals (shown in Figure 3 at 0m, 1m, 3m, 7m and 10m). Between stops, the barricade moves with a random velocity within the limits shown in Table I. The robot must use the mobile barricade for cover to reach the safety of the stationary barricades near the goal to efficiently navigate to the goal with as little exposure as possible to the observer. To achieve this, it has to recognize the mobile barricade as offering opportunistic and significant cover, and reactively follow in its shadow despite its time-varying velocity and stationary periods. Although it appears that the barricade provides ample cover for the robot, the 'shadow' width produced varies with the angle of the barricade relative to the observer. Consequently, at the beginning of the robot's traverse, it offers very little cover. This is demonstrated by the robot's position in the shadow of the mobile barricade in Figure 1.

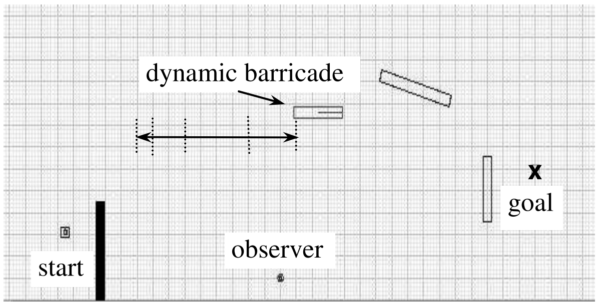

Figure 3. The environment configuration showing important locations, barricades, the mobile barricade's path and its stopping locations (broken lines).

The simulated robot has the dynamics of a two-dimensional Segway RMP in the sense that it can make a zero-radius turn but the sensors are not subject to the pitching platform. The parameter settings for the robot and the mobile barricade are shown in Table I. The robot's frontier timeout is set arbitrarily longer than the barricade's stationary time period. The distance between the safety zone and its shadow frontier is set to 0.8m and the width of the safety zone is 1.4m. The robot has accurate localization and uses a simulated SICK laser rangefinder for detecting and mapping objects in the environment to the occupancy grid. The laser is configured for 8m and provides a 180-degree scan of the environment in front of the robot.

TABLE I. PARAMETER VALUES USED FOR SIMULATION

	Robot	Mobile Barricade
Velocity	0.7 m/s (max.)	0.1-0.6 m/s
Timeout	15 s	10 s
Length	0.5 m	3 m

A. Results

The experiment was conducted 10 times; an example of a completed stealthy traverse is shown in Figure 4 as the robot's path on the potential field. Also shown is a typical path taken for navigating directly to the goal.

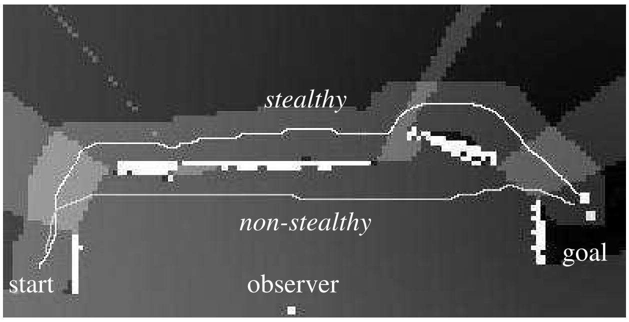

Figure 4. An example of stealthy and non-stealthy traverses shown on the global potential field. The irregular white objects are the barricades. Dark areas represent shadows behind the barricades and therefore attractive locations for the robot. The grey area surrounding the stealthy path is the accumulated effect of a local high-valued potential field that is positioned at the robot's location during the traverse to prevent waypoints being selected too close to it. The robot assumes that historical objects in the occupancy grid persist in the absence of new sensor data, hence the mobile barricade's appearance as a long object. This effect could be removed if the robot knew the size of the barricade or had omni-directional sensors.

Initially, while the robot was waiting at the first stationary barricade, the mobile barricade came into view. This produced a better opportunity for the robot, which then navigated to a position in the shadow of the mobile barricade. The robot effectively tracked in the barricade's shadow to maintain its position in the safety zone for the duration of the barricade's traverse, regardless of its movements. Since the barricade's timeout was less than the robot's, a frontier timeout only occurred when the robot was stopped at the stationary barricades.

To analyze the effectiveness of the approach, the traverses are evaluated by distance (efficiency) and the time the robot was visible to the observer (stealth). These results are compared to direct navigation traverses to the goal and the performance of our previous algorithm for static environments. These are annotated as 'Direct' and 'Static Stealth', respectively, in Table II.

TABLE II. AVERAGE SIMULATION RESULTS

	Distance (m)	Time (s)	Assumed Detected (s)	Actual Detected (s)
Direct	36.7	51	n/a	47
Static Stealth	46.5	60	35.1	39.2
Dynamic Stealth	45.1	121	16.2	18.9

The 'Distance' and 'Time' columns denote the total distance and time taken for the robot to travel from the start location to the goal. The 'Assumed Detected' column is the time the robot assumed it was being observed. This reflects the robot's internal analysis of the situation, which is prone to inconsistent representations due to sensor noise. The 'Actual Detected' column represents the actual amount of time the robot was detected by the observer.

It is obvious that the stealthy traverses produced higher stealth and lower efficiency than a direct traverse. However, the inability of the Static Stealth approach to fully utilize the mobile cover offered by the dynamic barricade lead the robot to outpace the barricade and make the majority of its traverses exposed to the observer. The travel times for the Dynamic Stealth traverses are lengthy since they include the mobile barricade's stopping times and the robot's timeouts at stationary barricades.

V. Real Robot Experiments

The experiments conducted in the real environments validate the simulation results. Whereas the simulation experiments demonstrated the algorithm's dynamic and opportunistic features using a barricade with a varying velocity, the experiments conducted in the real environment use a barricade that is initially located near the robot's start location and travels at a constant velocity between stopping locations.

The environments consisted of two separate grassy areas measuring approximately 35m by 25m. The experiments were conducted three times with a Pioneer AT and a Segway RMP using the parameters listed in Table III. Demonstration on different platforms indicates the generality of the approach.

TABLE III. PARAMETER VALUES FOR THE REAL ROBOT EXPERIMENTS

	AT / RMP	Mobile Barricade (AT / RMP)
Velocity	0.7 m/s	0.3 m/s
Timeout	7 s / 15 s	5 s / 10 s
Length	≈0.5 m	2.25 m

The dynamics of the two robots vary significantly. The Pioneer AT (right image in Figure 5) is a skid-steered four-wheel-drive robot with a limited turning radius while moving and a maximum velocity of 0.7 m/s. The Segway RMP (left image in Figure 5) is a two-wheel drive robot with zero-radius turning capability and can travel up to 3.5 m/s. It was limited to 0.7 m/s in the experiments for consistency. Its dynamics are based on an inverted pendulum controller so it pitches to move forward or backward. This consequently tilts the sensors whose readings were adjusted to compensate.

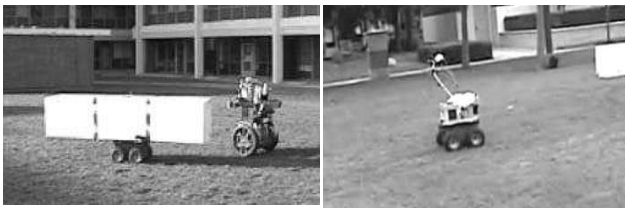

Figure 5. The Segway RMP and the mobile barricade (left) and the Pioneer AT (right) during a traverse.

Each robot carried a sensor suite consisting of a Garmin 16A GPS unit used in conjunction with a 3DMG IMU and the robot's odometry that provided localization accurate to within 2m. Each also carried a SICK laser rangefinder configured for 8m for obstacle avoidance and for mapping environment objects into their occupancy grid representation. The distance between the safety zone and the shadow frontier was set to 1.1m and the width of the safety zone was 1.4m for all experiments.

The barricade configuration was similar to the simulation experiment except the mobile barricade's start location. The mobile barricade consisted of three boxes fixed to a Pioneer AT base as shown in Figure 5. It travels for 10m stopping at 1m, 3m, 7m and 10m. Its parameters for the experiments are shown in Table III.

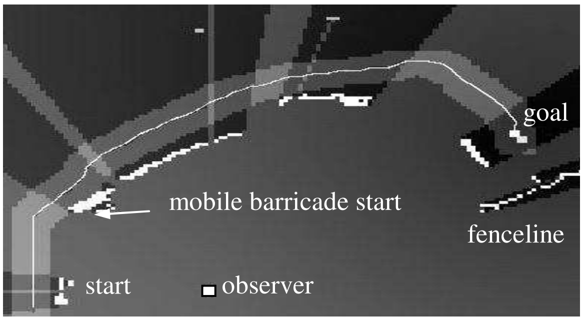

Figure 6. An example potential field from an AT traverse.

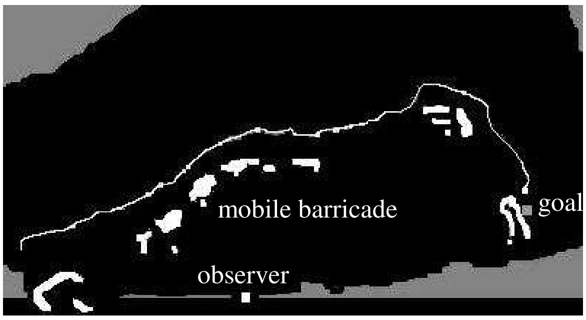

Figure 7. An example of the occupancy grid generated during an RMP traverse.

Examples of completed robot traverses are shown in Figure 6 and Figure 7 for the AT and RMP, respectively. Inconsistencies in the barricades' representations are a result of sensor noise overwriting the occupancy grid. This was more prevalent for the RMP.

TABLE IV. AVERAGE REAL ROBOT RESULTS

	Dist. (m)	Time (s)	Assumed Detected (s)	Actual Detected (s)
AT Direct	42.3	61	n/a	51
AT Stealth	54.9	154	38	32
RMP Direct	31.7	55	n/a	48
RMP Stealth	49.9	193	37	29

The average statistics for each experiment are shown in Table IV. Consistent with the simulation annotation, 'Time' and 'Dist.' are the robot's reported time and distance traveled. 'Assumed Detected' refers to the amount of time the robot assumed it was being detected. The actual detection times were determined from video taken from the observer's location. The discrepancy between the assumed and actual detection times are a result of sensor noise making the robot erroneously believe it was exposed or obscured at times. In each environment, direct navigation traverses to the goal (annotated as '<robot type> Direct' in Table IV) were conducted for comparison.

VI. MODEL-BASED ANALYSIS

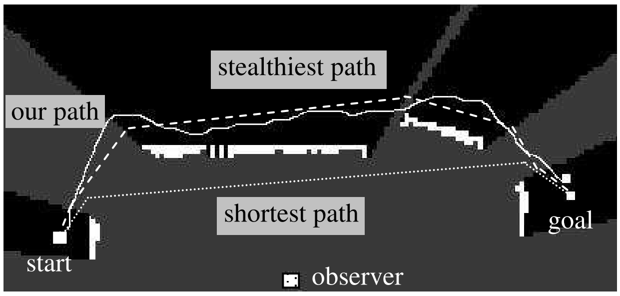

Figure 8. A potential field view with the most efficient (lower broken line), stealthiest (upper broken line) and our approach's (solid line) paths superimposed. Exposed/observable areas are grey and hidden areas are black.

While the simulation and real robot results indicate that our stealthy navigation algorithm achieves low visibility to an observer, it does not provide a qualitative indication. In this section, we compare our approach with those determined as providing empirically optimal stealth and efficiency for a given environment.

The comparison is made to a completed stealthy traverse from simulation as the control case. This traverse is typical for the robot in an *unknown* environment and is the base image in Figure 8. The comparison approaches are derived from the resultant occupancy grid generated by this traverse. In this analysis, the focus is on the path through the *known* environment with the dynamic barricade considered as an extended static object.

The most *efficient path* is simply defined as the shortest achievable path to the goal. This is determined by selecting a waypoint closest to the end of the barricade at the start location that would allow the robot to traverse safely around it. The same determination is made at the goal barricade. The resulting path is shown by the lower broken line in Figure 8.

The *stealthiest path* is determined by selecting waypoints on frontiers that represent the shortest distance through the observable areas to the goal location. Once again, this is offset from the barricades to represent a realistic obstacle-free traverse. The resulting path produces the minimum achievable observability and is shown by the upper broken line in Figure 8.

The example for our approach represents typical decisions the robot makes through an environment based on the parameter values chosen for the potential fields. Its distance measurement was derived using the straight line distances between each point where the path crossed a frontier. This removes the wandering effects of the robot's navigation (mainly when tracking the moving barricade) which generates longer paths, and provides a fair comparison to the derived paths.

1) Evaluation

Each path in this analysis is evaluated according to its stealth and efficiency relative to the stealthiest (*p_stealth*) and most efficient (*p_efficient*) paths above. A completed path p consists of an observable part p_{obs} and a hidden part p_{hid} which represents the length of the path that was exposed to and hidden from the observer respectively. The criteria for determining a paths' stealth is based on comparing its observability to *p_stealth* as shown in (1). This is a more intuitive measure than comparing the lengths of the paths obscured in shadows since they could be arbitrarily long.

$$Stealth = \text{length}(p\_stealth_{obs})/\text{length}(p_{obs}) \quad (1)$$

A path's efficiency is simply a comparison of its length compared to *p_efficient* as shown in (2).

$$Efficiency = \text{length}(p\_efficient)/\text{length}(p) \quad (2)$$

Together, these stealth and efficiency metrics provide evidence of the path's quality. The results of applying each metric to the *p_stealth*, *p_efficient*, and our algorithm's paths are shown in Table V.

TABLE V. ANALYSIS OF PATH QUALITY

	Efficiency	Stealth
Shortest Path	100%	36%
Stealthiest Path	86%	100%
Our Approach	84%	91%

The results indicate that our approach provides a high quality stealthy path according to the evaluation function. This is encouraging considering that this analysis is conducted with a known environment and the traverse from our approach was from an unknown environment. Stealth occurs from the potential fields effectively pushing the robot's path away from the observer to behind the barricades. The robot chooses an efficient path as a result of being attracted to the goal and to the shadows. The combination of these effects rationalize using potential fields to encode the stealth-efficiency cost function and as an intuitive approach for reactive stealthy navigation.

VII. SUMMARY

We have presented an approach for stealthy reactive navigation in unknown dynamic environments in the presence of an observer. For a robot to be stealthy under these constraints, it needs to take advantage of coverage opportunities as they occur, particularly since a dynamic object offering beneficial cover may only persist briefly. Our algorithm is a frontier-based method that allows the robot to reactively hide behind a mobile object and dynamically adjust its position according to the movement of the object. Thus the method is capable of handle varying object velocities or even objects that stop for limited time periods. As such, our approach makes very few assumptions as to the nature of the dynamics of the objects. If the objects traveled in repeatable patterns, higher level assessments of them could be made and models created to assist the robot with stealthier subsequent traverses. It cannot be assumed that objects with these characteristics will be present in an unknown environment and hence a more reactive approach provides a higher utility in general. We have validated and evaluated our reactive approach in simulation and outdoor environments using an object with varying dynamics. High quality stealthy efficient paths are produced that compare well with empirically-determined optimal solutions generated through offline analysis.

REFERENCES

[1] A. Tews, G. Sukhatme, and M. Matarić, "A multi-robot approach to stealthy navigation in the presence of an observer," Proceedings of the International Conference on Robotics and Automation, New Orleans, LA, Apr. 26 – May 01, 2004, pp. 2379-2385.

[2] Y. Teng, D. DeMenthon, and L. Davis, "Stealth terrain navigation," IEEE Transactions on Systems, Man, and Cybernetics, vol. 23, no. 1, 1993, pp.96-110.

[3] S. Ravela, R. Weiss, B. Draper, B. Pinette, A. Hanson, and E. Riseman, "Stealth navigation: planning and behaviors," ARPA Image Understanding Workshop, Monterey, CA, 1994, pp. 1093-1100.

[4] S. Bortoff, "Path planning for UAVs," Proceedings of the 2000 American Control Conference, Chicago, IL, vol. 1, no. 6, 2000, pp. 364-368.

[5] D. Cook. P. Gmytrasiewicz, and L. Holder, "Decision-theoretic cooperative sensor planning," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 18, no. 10, 1996, pp.1013-1023.

[6] E. Birgesson, A. Howard, and G. Sukhatme, "Towards stealthy behaviors," IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, Oct. 2003, pp. 1703-1708.

[7] A. Elfes, "Occupancy grids: a probabilistic framework for robot perception and navigation," PhD Thesis, Carnegie Mellon University, 1989.

[8] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," The International Journal of Robotics Research, vol. 5, no. 1, 1986, pp. 90-98.

[9] R. Arkin, "Integrating behavioral, perceptual, and world knowledge in reactive navigation," Robotics and Autonomous Systems, no. 6, 1990, pp. 105-122.

[10] B. Gerkey, R. Vaughan, K. Støy, A. Howard, G. Sukhatme, and M. Matarić, "Most valuable player: a robot device server for distributed control," Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2001), Maui, Hawaii, 2001, pp. 1226-1231.