

# Autonomous Airborne Geomagnetic Surveying and Target Identification

Christopher W. Lum,<sup>\*</sup> Rolf T. Rysdyk,<sup>†</sup> and Anawat Pongpunwattana <sup>‡</sup>

*Autonomous Flight Systems Laboratory*

*University of Washington, Seattle, WA, 98105, USA*

This work considers algorithms for maritime search and surveillance missions. Search and identification of magnetic anomalies are evaluated. A combination of a particle filter and a neural network are used to identify and classify anomalies. Communication among vehicles is assumed to update a centralized occupancy based map which represents a discretized belief of target locations. Control decisions are based on a nearest neighbor search of the surrounding cells of the occupancy map. Simulation is performed using a planar kinematic model and actual aeromagnetic data.

## Nomenclature

$\alpha$	Heading penalty coefficient
$\Delta\psi$	Change in heading
$\sigma$	Standard deviation
$\chi$	Particle filter set
$\psi$	Heading
$B$	Score of occupancy map cell
$C$	Confidence measure of particle filter
$\overline{D}$	No detection event
$e_i$	$N \times 1$ vector of zeros with 1 in $i^{th}$ position
$f()$	Gaussian distribution
$g()$	Sampling function
$h()$	Target magnetic signature function
$J$	Utility function
$M$	Total number of particles
$p(A B)$	Conditional probability of $A$ given $B$
$\bar{u}$	Control vector
$V_a$	Airspeed
$w$	Weight on particle
$\bar{x}$	State of agent with respect to target
$z$	Sensor measurement

### *Subscript*

$t$	Time index
$tgt$	Target
$uav$	Agent (UAV)

### *Superscript*

$[k]$	Agent (UAV) number index
-------	--------------------------

<sup>\*</sup>Research Assistant, Dept. of Aeronautics and Astronautics, lum@aa.washington.edu, AIAA student member

<sup>†</sup>Assistant Professor, Dept. of Aeronautics and Astronautics, rysdyk@aa.washington.edu, AIAA member

<sup>‡</sup>Research Associate, Dept. of Aeronautics and Astronautics, anawatp@u.washington.edu, AIAA member

$[m]$	Particle number index
$tgt$	Target frame of reference
$uav$	Agent (UAV) frame of reference
*	True measurement or state

## I. Introduction

Maritime search and surveillance-type missions typically require heavy human involvement. A set of unmanned aerial vehicles (UAVs) has the potential to provide a “sensor network” to greatly increase efficiency and effectiveness of these surveillance type missions. However, limited autonomy is still a bottleneck to networked UAV applications. High operator involvement is required in logistics and operation, for example, distributing assignments such as which regions to search and coordinating subsequent sensor measurements. In a noisy environment, it becomes difficult for a human operator to classify sensor readings and assign confidence in these readings. Determining regions of high target-location probability and coordinating nearby agents to converge on a particular spot while allowing other vehicles to continue searching is also difficult. Therefore, the primary limitation to concurrent operation of multiple vehicles remains lack of autonomy of these vehicles.

Advances in miniaturized electronics, Global Positioning System (GPS) technology, and sensors (video and infrared cameras, magnetometers) coupled with sophisticated guidance, navigation and control algorithms, enable the development of small UAVs for Intelligence, Surveillance and Reconnaissance (ISR) missions operating for extended periods of time over large geographical areas. The development of small inexpensive UAVs will allow a flexible and robust distributed sensor network to replace limited manned flights or large UAVs that concentrate expensive sensing and communication systems in a single agent with a large team of operators.

Adaptive real-time mission planning algorithms on-board small UAV platforms will give the vehicles greater autonomy, significantly reducing operator requirements, and thereby enable the deployment of sensor networks. Algorithms for solving real-time task and path planning (TPP) problems were developed at the University of Washington.<sup>1</sup> These algorithms were integrated and evaluated with target observation and tracking algorithms and were shown to have significant promise for real-time on-board implementation for a range of ISR missions.<sup>2</sup> The current work aims to develop similar algorithms for the Georanger UAV<sup>3</sup> for aero-magnetic surveying. A specific application is the detection of a submarine in littoral waters based on its magnetic signature. Eventually, these algorithms would be developed to operate with a team of heterogeneous vehicles. This team would be comprised of individual vehicles known as agents. Each agent could have different capabilities and sensors which dictates that algorithms be easily adaptable to accommodate these differences.

Objectives in this work include: evaluation of a search strategy with submarine magnetic models; integration into the task and path-planning architecture;<sup>1</sup> hardware-in-the-loop implementation and evaluation;<sup>4</sup> implementation on board the Georanger UAV;<sup>5</sup> communications relay logic between multiple Georangers and one or more ground stations; flight testing and evaluation of performance.

This work contributes to the large area of research in autonomous target identification and searching algorithms. Fox, et al.<sup>6</sup> have been involved in state estimation and robotic localization. Their work is closely related to the particle filter method which is used here for identification. Other groups such as Durrant-Whyte et al.<sup>7</sup> and Polycarpou et al.<sup>8</sup> have developed methods and algorithms for actively searching a region for a target using teams of autonomous agents.

Section II introduces the concept of a local total magnetic intensity (TMI) map. The use of this map in searching for magnetic anomalies is also explained. Once an anomaly is encountered, the problem of identifying and classifying this anomaly using a particle filter is covered in Section III. Once the target is identified, the occupancy map of the area is updated and guidance decisions are made by evaluating a utility function, described in Section IV. Finally, conclusions and continuing research directions are presented in Section V.

## II. Aeromagnetic Data Surveys

### A. The Georanger autonomous aeromagnetic survey vehicle

For aeromagnetic surveys, the agent (UAV) is essentially a mobile sensor. The vehicle autonomy serves the engineering user who simply specifies an area of interest and, after some processing, receives a corresponding set of data. In this idealized perspective, the data-analyst is unconcerned with the method with which the data was obtained. To achieve such an objective, autonomy is required at several hierarchical levels. The autonomy of decision making and task distribution is addressed in earlier work.<sup>1</sup> The current work focuses on target detection and search and surveillance tactics. The vehicle serving as the sensor platform in this work is the Fugro Georanger, provided by The Insitu Group, and is shown in Figure 1.

The Georanger is a derivative of the Seascan vehicle which has the following performance specifications:

Max Takeoff Weight	41.9 lb / 19 kg
Payload	15.4 lb / 7 kg
Endurance	15 hours
Service Ceiling	16400 ft / 5000 m
Max Level Speed	70 knots / 36 m/s
Cruise Speed	49 knots / 25 m/s
Wing Span	10.2 ft / 3.1 m
Fuselage Diameter	7.0 in / 0.2 m
Length	4.9 ft / 1.5 m

Each agent is equipped with a magnetometer to measure the total magnetic intensity at its current location. This data is relayed to a ground station. The crucial piece of information required by the ground station is a local magnetic map of the region where the search is taking place. This map of the TMI of the region may be acquired using analytical models such as the WMM-2000 or WGS-84 model.<sup>9</sup> However, since these models are coefficient-based analytical models, they do not capture temporal or small local variations in magnetic field strength. Therefore, a more accurate map is obtained by performing an actual survey over the area of interest to collect the necessary data.

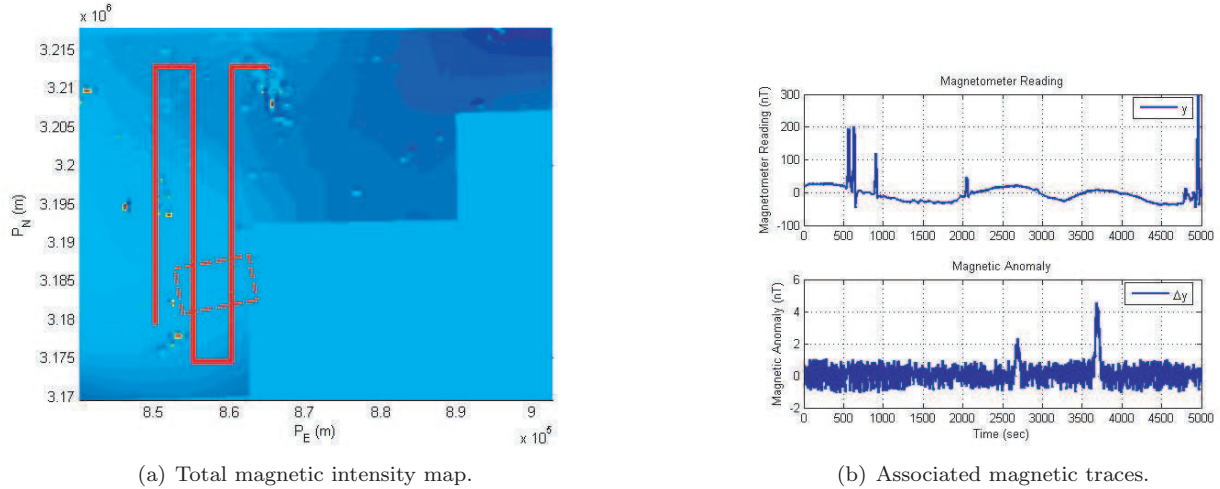


Figure 1. Image of the Fugro autonomous aeromagnetic survey vehicle, the 'Georanger I' by the Insitu Group.

### B. Total Magnetic Intensity maps

When an actual search is executed, differences between the ground station map of the magnetic field and the actual magnetic field will appear as magnetic anomalies. In this work, to minimize the number of false anomaly encounters and to increase the accuracy of the evaluation, actual magnetic survey data is used as a local TMI map. This data is provided by Fugro Airborne Surveys. The data was collected by a manned

aircraft equipped with a magnetometer to measure the TMI. This information, coupled with a GPS position, provides the TMI in “line data” form. This data can then be interpolated into a 100x100 meter grid. TMI readings at locations other than survey points are linearly interpolated from this grid. A magnetic map of a region in the Gulf of Mexico and a simple grid search trajectory are shown below in Figure 2(a). Here, the data is acquired in an approximate 60x50 km grid. The regions of uniform color denote areas where survey data is not available, creating the “staircase” appearance. Assuming that there are only permanent fixtures in the region when the map is acquired, this map now makes up the reference set of data on the ground station.



**Figure 2.** The total magnetic intensity map and trajectory over area with corresponding magnetometer readings.

In addition to a local magnetic map, a magnetic model of the desired target is also required. In the following example, the magnetic signature of the target (an idealized submarine) is modeled as a simple two dimensional Gaussian distribution, shown in Figure 3. The magnetic signature of the target is a function of many variables, namely depth of target, sensor altitude, etc. For current purposes, the target is assumed stationary and at a fixed depth, thereby rendering the magnetic signature static. Assuming that the magnetic signature of the target simply adds to the total magnetic intensity of the local region in a linear fashion, anomalies can easily be identified by simply subtracting the magnetometer reading from the local reference map which is stored on the ground station.

The described approach can be used to compare magnetometer readings with the reference data to create a differential measurement. Large differential measurements imply the presence of a new magnetic anomaly and possible target. If the agent does not fly over any targets, the magnetic anomaly should be near zero. Small non-zero anomaly encounters can be attributed to temporal variations in magnetic field and sensor noise. A simple grid search pattern was shown previously in Figure 2(a). The location of the target is shown as a dashed red box and the trajectory of the agent is shown in the solid red line (starting in the lower left corner). The associated total magnetic intensity trace and differential measurement trace is shown in Figure 2(b). The total magnetic intensity reading as the agent flies over this trajectory is shown in the upper trace and the differential measurement is shown in the lower trace. As the agent flies this search trajectory, the sensor measurement is constantly compared to the reference data set to generate a differential measurement. As can be seen in Figure 2(b), given the differential magnetometer reading, it is obvious how to detect where the anomaly occurred (two spikes at approximately 2700 and 3700 seconds) even though the actual range of absolute measurements may be large.

### III. Identifying Anomalies Using Particle Filters

Magnetic anomalies can be caused by many factors such as temporal variations in the magnetic field or false targets encounters (i.e. boats/vessels). Once a magnetic anomaly is encountered, it must be identified and classified. On simplistic level, the overall goal is to either classify the anomaly as the desired target or a false reading. Obviously, it would be simple to identify the anomaly if the entire magnetic signature of the

anomaly is obtained (the UAV flies over the entire boxed region in Figure 2(a)). However, this requires many passes over a potential target, and significant time to make the necessary measurements. If the anomaly is moving or evading, this may not be feasible. The question now becomes, given only one or two passes over the target, is it possible to correctly identify or provide a probability that this anomaly is indeed the target being sought after? To address this issue, a particle filter method is used.

A particle filter is a recursive, non-parametric Bayes filter technique which estimates the states of a system using a finite number of state hypotheses.<sup>10</sup> In this situation, the state vector that is being estimated is the position of the agent with respect to the target, expressed in the target's frame of reference and the relative heading of the agent with respect to the target.

$$\bar{x}_t^{[m]} = \begin{bmatrix} x_{uav/tgt}^{tgt} \\ y_{uav/tgt}^{tgt} \\ \psi_{uav/tgt} \end{bmatrix} \quad (1)$$

Each individual state hypothesis,  $\bar{x}_t^{[m]}$ , is referred to as a particle, and together they make up the particle filter set,  $\chi_t$ .

$$\chi_t = \bigcup_M \bar{x}_t^{[m]} = \left\{ \bar{x}_t^{[1]}, \bar{x}_t^{[2]}, \dots, \bar{x}_t^{[M]} \right\} \quad (2)$$

GPS allows the position of the agent in the earth frame to be computed, but the target location and orientation in the earth frame is not known. The goal of the particle filter is to estimate the state of the agent (position and orientation with respect to the target, expressed in the target's frame of reference). The true location of the agent with respect to the target expressed in the target's frame of reference at a time  $t$  is denoted as  $\bar{x}_t^*$ . The particle filter performs this estimate using two main steps, a prediction and correction step.

### A. Prediction

In the prediction step, each particle is propagated forward in time using a motion model of the individual agent.

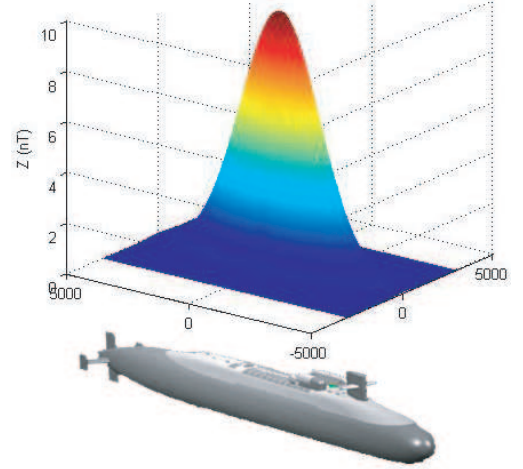
$$\bar{x}_t^{[m]} = g \left( p \left( \bar{x}_t^{[m]} | \bar{u}_t^*, \bar{x}_{t-1}^{[m]} \right) \right) \quad (3)$$

In Eq. (3),  $g$  is a sampling function which simply chooses a sample from a probability density function. Each new particle is created from the old particle and the current control (applied to transition particle at time  $t - 1$  to time  $t$ ). The term  $p \left( \bar{x}_t^{[m]} | \bar{u}_t^*, \bar{x}_{t-1}^{[m]} \right)$  is a multi-dimensional probability density function of the new state given the old state and current control. Notice that in this formulation, the state transition is not a deterministic process. This stochastic aspect actually has important implications regarding the robustness of the particle filter.<sup>10</sup>

Although  $p \left( \bar{x}_t^{[m]} | \bar{u}_t^*, \bar{x}_{t-1}^{[m]} \right)$  may be difficult to compute analytically, Eq. (3) is implemented in simulation by simply adding noise to the control and then propagating the state forward using a deterministic motion model (a simple kinematic model in this case). The control vector for the model is simply

$$\bar{u}_t = \begin{bmatrix} V_a \\ \Delta\psi \end{bmatrix} \quad (4)$$

In simulation, the noise added to each element of the control vector is obtained by sampling from a normal, Gaussian distribution with a variable standard deviation,  $\sigma$ . The standard deviation is a function of the actual control applied to the agent,  $\bar{u}_t^*$ . In effect, as  $\|\bar{u}_t^*\|$  increases, so does  $\sigma$ . Physically, this translates into a model whose state transition becomes more uncertain as the agent moves faster or executes larger heading changes.



**Figure 3. Magnetic signature of target. Magnetic signature given by  $z = h(x, y)$ .**

In addition to the control input at each time step, the actual sensor measurement observed by the agent,  $z_t^*$ , is made available to the particle filter. Each particle is then assigned a weight,  $w_t^{[m]}$ , based on how likely it is to make the same sensor measurement at its current state.

$$w_t^{[m]} = p\left(z_t^* | \bar{x}_t^{[m]}\right) \quad (5)$$

In effect, this should assign high weights to particles whose states are close to the actual state,  $\bar{x}_t^*$ . Notice that Eq. (5) does not require a sampling function like Eq. (3) because  $z_t^*$  and  $\bar{x}_t^{[m]}$  are known at this point. Eq. (5) describes the sensor model of the agent. It allows for the fact that even though a particle's state may be vastly different than the true state of the agent, if the sensor is poor or unreliable, it has the possibility of still making the same sensor reading as the agent.

The sensor model used in simulation calculates Eq. (5) by creating an error between the particle sensor measurement and the true sensor measurement and then using this as the argument of a normal, Gaussian distribution.

$$w_t^{[m]} = f\left(z_t^* - z_t^{[m]}\right) \quad (6)$$

In Eq. (6),  $z_t^{[m]}$  is the predicted sensor measurement made by particle  $m$ . In simulation,  $z_t^{[m]}$  is only a function of the first two states and is generated using the target's magnetic signature function (Figure 3) to obtain  $z_t^{[m]} = h(x_{uav/tgt}^{tgt}, y_{uav/tgt}^{tgt})$ . Furthermore,  $f$  is a Gaussian distribution with zero mean and standard deviation  $\sigma$ . As stated previously,  $\sigma$  can be adjusted based on the sensor model. A larger  $\sigma$  implies an unreliable sensor; therefore, particles that do not make the same measurement as the true agent still receive high weights. Note that the weight is not a probability, but this still achieves the goal of assigning high weights to particles that are more likely to have states which are similar to the true agent state.

The majority of this section has discussed the state estimation problem. Historically, particle filters have been employed in this manner to perform tasks such as localization<sup>11</sup> and state estimation.<sup>6</sup> These are certainly important tasks in this problem; however, to perform target identification, a closer look at the weights,  $w_t^{[m]}$ , is warranted.

A scalar quantity which collectively measures the overall accuracy of the particle filter can be obtained by simply summing all the weights. If most of the particles are in locations that are similar to the true state, then the sum of the overall weights should be large. Traces of  $C_t$  for two different situations are shown later in Figure 6.

$$C_t = \sum_{m=1}^M w_t^{[m]} \quad (7)$$

This trace of a  $C_t$  vs.  $t$  might be considered a side-effect of estimating  $x_t^*$ , but as will be shown later, this is the main piece of information that will be used to address the target identification problem.

## B. Correction

Now that each particle has been propagated forward and assigned a weight, it becomes necessary to correct the particle filter set so that it comes closer to representing the actual state of the agent. This process is known as resampling.

As stated before, the particle filter's estimate of the state is made up of all the particles. Currently, the particle filter set contains particles which have both high and low weights. As more and more sensor measurements are acquired, it is desired that high scoring particles are replicated and kept in the next generation population whereas low scoring particles are discarded. The important feature in this evolutionary process is that the particles are resampled with replacement so that the total number of particles remains constant at each cycle. Any type of evolutionary scheme, such as survival of the fittest, can be used to evolve the current population to the next.

In simulation, a roulette wheel method is used. In this method,  $M$  bins are created (one for each particle). The size of each bin is directly proportional to the weight of the associated particle. The bins are placed next to each other and a random number is then generated. The bin in which the random number falls then has its associated particle included in the next population. This process is repeated  $M$  times and is



synonymous to spinning a roulette wheel  $M$  times where the number and size of the slots on the wheel are directly proportional to  $M$  and the weights, respectively.

Using the roulette wheel method yields resampling proportional to the weights. This allows for a particle to be copied multiple times in the next generation. This also generates a small probability that particles with low weights have the possibility to survive to the next generation as well.

One important feature of the particle filter is the ability to use different motion and sensor models. This allows for a team of agents to be comprised of different types of vehicles and sensors. This simply requires modifying the motion and sensor models of each particle filter for each member of the heterogeneous team.

### C. Execution

When an agent encounters an anomaly whose magnitude exceeds the noise threshold (approximately 1 nT in this case), the particle filter is started in an attempt to estimate the state of the agent with respect to the target. In addition to this, recall that the trace of  $C_t$  vs.  $t$  is the true product of the particle filter that is used for target identification. The particle filter's progression as the agent flies diagonally over the target is displayed over a top down view of the target signature (Figure 3) and is shown below in Figure 4.

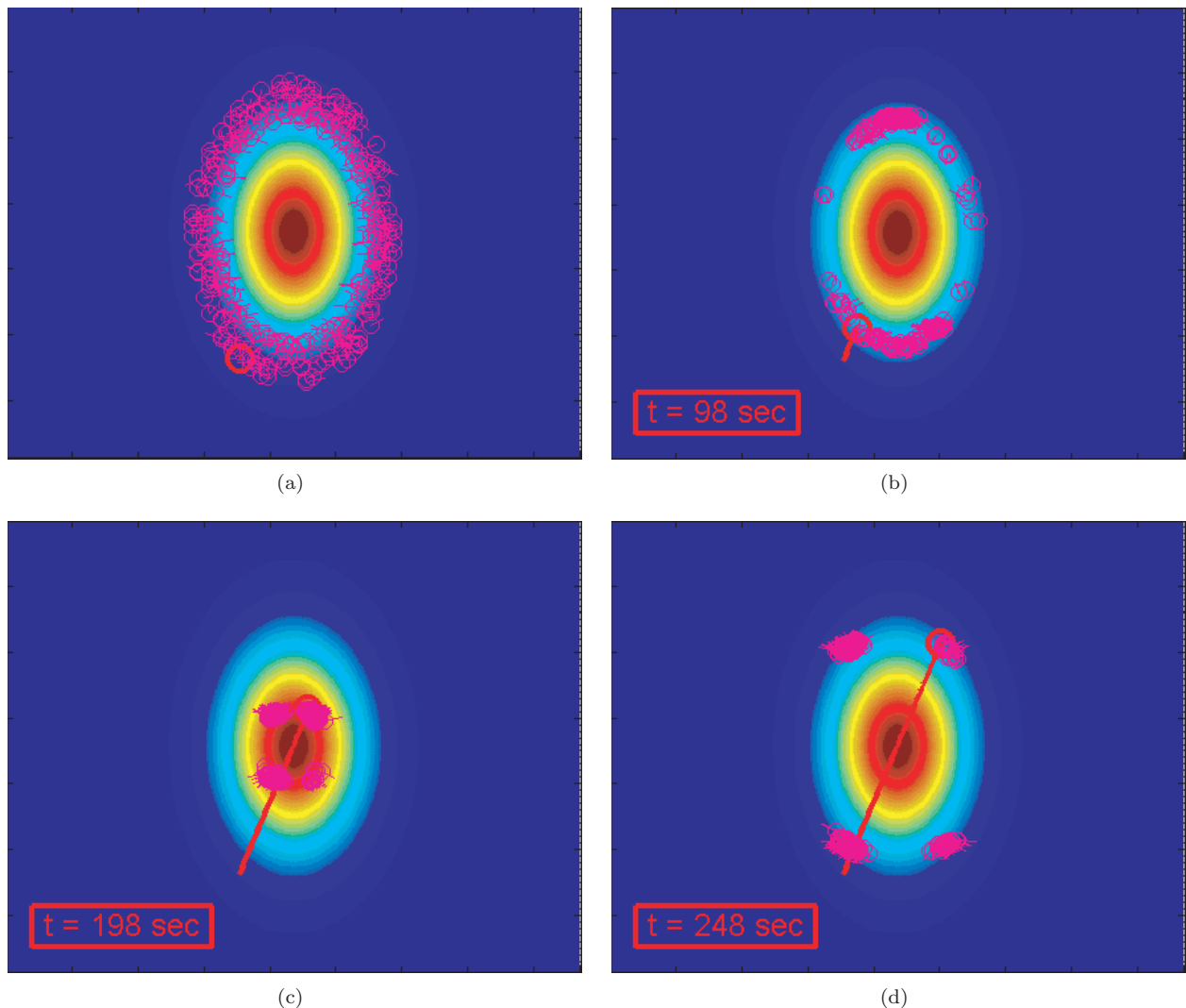


Figure 4. Particle filter progression during a target encounter. The solid line indicates actual aircraft position relative to target signature, while the particles concentrate about possible positions

In this sequence, the large red circle represents the actual location of the agent and the solid red line represents the agent's trajectory over the target. The smaller purple dots represent the particle filter's many

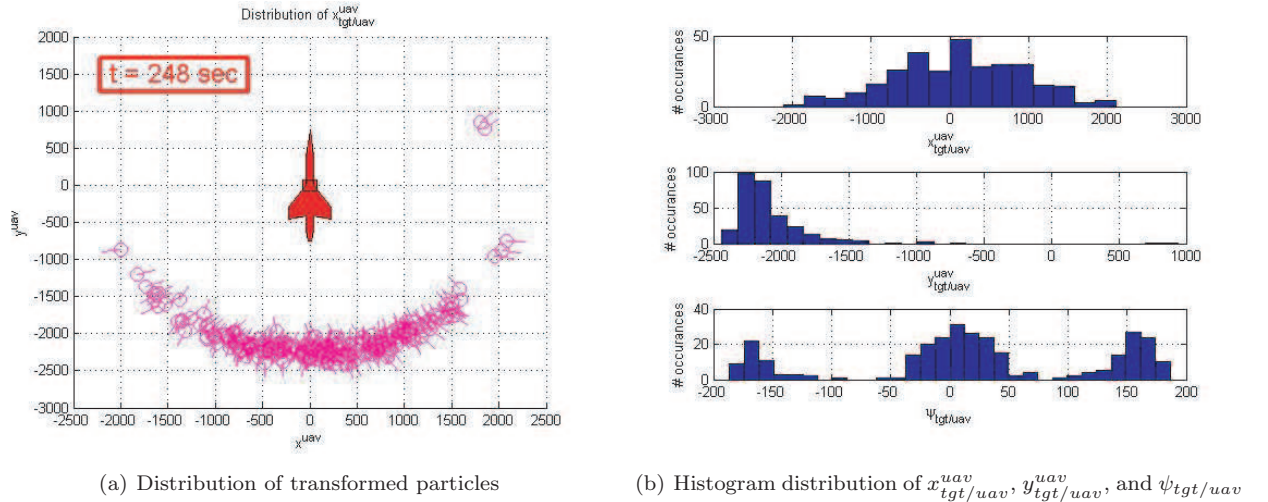
different hypotheses of the possible state of the agent (position north, position east, and heading). The actual agent crosses over the target starting in the lower left corner and flies over it to the upper right corner. Also note that the initial distribution of particles is not simply random over the domain. Since the algorithm is recursive, the number of iterations before convergence is based on its initial condition. Incorporating a priori knowledge that the particle filter is started when the anomaly magnitude exceeds 1 nT suggests that the particles be clustered along the level curves where the target signature is 1 nT.

As the agent obtains more and more sensor measurements (at a simulated rate of 1 Hz), the particle filter is able to eliminate particles which are inconsistent with the current measurement and resample these particles to regions which have a higher probability of producing the actual sensor reading,  $z_t^*$ . This is why as time progresses, the particles become concentrated around the actual UAV location. Near the end of the simulation, there are four distinct groups of particles. This is due to the symmetry of the underlying target signature. Each of these four groups of particles are equally likely because each group would produce the correct actual sensor readings. In effect,  $z_t^{[m]} \approx z_t^* \forall m$ . Because of this symmetry, the particle filter is not able to uniquely identify the position of the agent with respect to the target. This would require multiple passes over the target and more sensor measurements.

Although the goal of the particle filter is to estimate the position of the agent with respect to the target in the target frame of reference, in the larger picture, the location of the target with respect to the agent in the agent frame of reference is more useful because it then becomes simple to locate the target in the earth frame of reference (agent's position and orientation in the earth frame of reference is known from GPS). Each particle can be transformed using Eq. (8).

$$\begin{bmatrix} x_{tgt/uav}^{uav} \\ y_{tgt/uav}^{uav} \\ \psi_{tgt/uav} \end{bmatrix} = \begin{bmatrix} -\cos(\psi_{uav/tgt}) & \sin(\psi_{uav/tgt}) & 0 \\ -\sin(\psi_{uav/tgt}) & -\cos(\psi_{uav/tgt}) & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} x_{uav/tgt}^{tgt} \\ y_{uav/tgt}^{tgt} \\ \psi_{uav/tgt} \end{bmatrix} \quad (8)$$

When each particle is transformed in this fashion, the distribution of the target location with respect to the agent in the agent's frame of reference becomes as shown in Figure 5.



**Figure 5. Particles now represent position and orientation of target with respect to the agent in the agent's frame of reference.**

As shown in the first two plots in Figure 5(b), it appears that the particle filter now has a somewhat unique estimate of the location of the target relative to the agent as shown by an approximate unimodal distribution in  $x_{tgt/uav}^{uav}$  and  $y_{tgt/uav}^{uav}$  centered at approximately 0 and -2250, respectively. However, notice that the distribution of  $\psi_{tgt/uav}$  is obviously a multimodal distribution. This distribution is actually the sum of four peaks which should ideally be centered at  $\pm 11.3$  degrees and  $\pm 168.7$  degrees. Since the number of particles was not large enough and since the motion and sensor models of the particle filter were not highly accurate, the two peaks centered at  $\pm 11.3$  degrees appears as a single peak at 0 degrees.

This multimodal distribution in  $\psi_{tgt/uav}$  reflects the four distinct state hypotheses shown previously in Figure 4(d). However, if the orientation of the target is not desired, then by transforming the particles, it



is possible to obtain a unique estimate of simply  $x_{tgt/uav}^{uav}$  and  $y_{tgt/uav}^{uav}$ . Note that this is only the case when the agent happens to fly directly over the target as shown in this example. In a more general case where the agent passes over the target off-centered, then even with the transformation of the particles, the location of the target cannot be determined uniquely (but the number of possible locations may be reduced).

The previously described algorithm will perform regardless if the anomaly encountered is the actual target or a false anomaly. A method to identify the target is now required. The sum of all the particle weights,  $C_t$ , provides a qualitative measure of how confident the particle filter is that the anomaly encountered is the actual target. If all or most of the particles are resampled to areas which are near the actual state of the agent, then most of the weights will be fairly high. The sum of the particle weights for an encounter with the actual target and an encounter with a false anomaly is shown below in Figure 6.

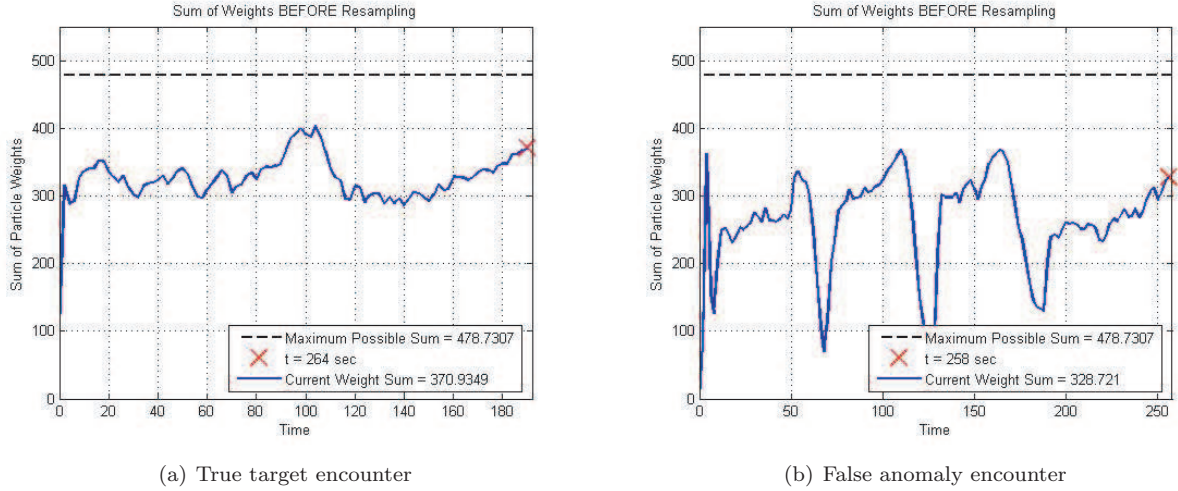


Figure 6. Sum of all particle weights during a true target encounter and a false anomaly encounter.

In Figure 6, the difference between a true target encounter and a false anomaly encounter is fairly clear. In the situation where the agent encounters the true target, the confidence measure increases initially as the particles are quickly resampled to locations which are consistent with the actual sensor measurements and then stays fairly constant. However, in the case where the agent encounters a false anomaly, the particle filter regularly “loses confidence” as inconsistent sensor measurements are obtained. This is characterized by the sharp drops in the sum of the particle weights. Current research is directed towards training a neural net to recognize these features and thus provide a qualitative measure to the target identification problem. In the end, the particle filter will provide the trace of the sum of the weights over time (Figure 6) and the neural net will process this trace. In combination, the particle filter and neural network provide a mapping from magnetic sensor measurements to a single scalar value which represents a measure of how confident the particle filter is that the encountered anomaly is the desired target or not.

## IV. Occupancy Map Based Searches

The particle filter method is used to identify anomalies once they are encountered. A method to actively search for targets and anomalies is now considered.

### A. Utility Function

As can be seen in Figure 2(a), one search pattern that can be used is a simple grid search pattern. However, for a team of autonomous agents, a more intelligent approach is desirable. In this situation, an occupancy based map search is employed. In this scheme, the search domain is discretized into rectangular cells. Each cell is assigned a score based on the probability that the target is located in that grid. This is similar to a two dimensional, discretized probability density function.<sup>7</sup> This centralized occupancy based map is shared and updated by all agents involved in the search. At each time step, guidance decisions for each agent are chosen based on this map. The individual utility function for each agent is given by

$$J^{[k]} = \bar{e}_i^T \bar{B}^{[k]} - \alpha^{[k]} \left| \psi_{curr}^{[k]} - \bar{e}_i^T \bar{\psi}^{[k]} \right| \quad (9)$$

The set  $\bar{B}^{[k]}$  is an  $N \times 1$  vector of occupancy map scores in the surrounding neighborhood of agent  $k$ . In current simulations, this population is only the eight cells immediately surrounding the agent (cells to northwest, to the north, to the northeast, etc.). This amounts to a 1-step-ahead predictor. This population size can be increased to encompass more cells at the cost of additional computational requirements. The second term is effectively a penalty on heading deviation.  $\psi_{curr}^{[k]}$  is the current heading of the agent and  $\bar{\psi}^{[k]}$  is an  $N \times 1$  vector of the heading required to orient agent  $k$  with each cell associated with the set  $\bar{B}^{[k]}$ . These angles and elements are shown in Figure 7.

The term  $\alpha^{[k]}$  is used to adjust the penalty associated with changing the heading of the agent. For example, if the agent is a large plane or boat, a heading change may be difficult and thus, a large  $\alpha^{[k]}$  would greatly penalize heading changes. This would result in an agent which would choose to possibly search a cell of lower score if it required a smaller heading change. In contrast, if the agent is a small UAV or helicopter with quick maneuvering capabilities, a very small  $\alpha^{[k]}$  would allow the agent to make frequent turns. As  $\alpha^{[k]}$  approaches zero, the agent follows a simple gradient climb algorithm ( $\alpha^{[k]}$  must be non-zero to avoid ambiguity in situations where several elements of  $\bar{B}^{[k]}$  have the same score).

For each agent, a control that maximizes Eq. (9) is applied. This amounts to choosing the correct value of  $i$  in Eq. 9. As the set  $\bar{B}^{[k]}$  becomes large and the utility function becomes more complex, it may be difficult to find an optimal solution in a single computational cycle. Research has been done concerning using evolutionary computation ideas to find a sub-optimal trajectory under a time constraint.<sup>12</sup>

## B. Updating Occupancy Map

Once the controls for each agent (i.e. a desired heading) are assigned, it becomes necessary to update the score of each cell based on the agent's findings. As an agent finishes searching a cell, if no anomaly is discovered, the score of the cell can be updated using the sensor model for each agent. If an anomaly is present, the probability that an agent detects this anomaly is given by  $p(D^{[k]}|\bar{x}^{[k]}, \bar{x}_{tgt})$ . Therefore, the probability that the agent will "miss" detection is simply the complement.<sup>7</sup>

$$p(\bar{D}^{[k]}|\bar{x}^{[k]}, \bar{x}_{tgt}) = 1 - p(D^{[k]}|\bar{x}^{[k]}, \bar{x}_{tgt}) \quad (10)$$

The score of each cell in the event of not detecting an anomaly is now updated using

$$B_{i,new}^{[k]} = B_{i,old}^{[k]} p(\bar{D}^{[k]}|\bar{x}^{[k]}, \bar{x}_{tgt}) \quad (11)$$

In the event of an anomaly encounter, the particle filter is used to identify and classify the anomaly. If it is determined that it is not the target, Eq. (11) can simply be used to update the cell. If the anomaly is determined to be the target, then the surrounding cells of the occupancy map within a certain radius have their scores increased. The radius chosen can be a function of the target identification algorithm's confidence that the anomaly is indeed the target. This allows for a local increase in belief of the target position while leaving other areas unaffected.

The formulation of the utility function and the occupancy map update also caters to the theme of heterogeneity that was introduced with the particle filter. Both the utility function and occupancy map update equation can be adjusted to accommodate for a heterogeneous team of agents.

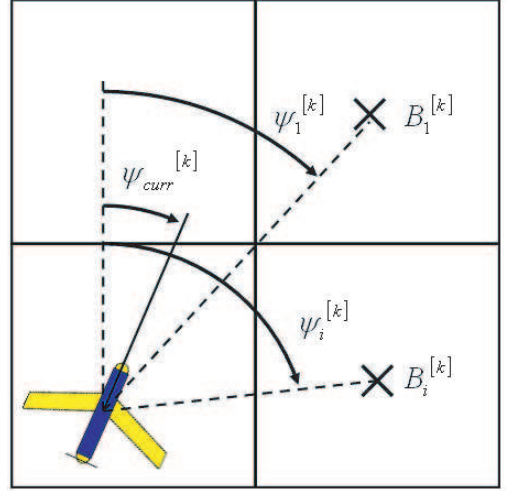


Figure 7. Angles and scores in occupancy map.

### C. Execution

In the following example, a simple two dimensional Gaussian distribution is used to increase the scores over the two dimensional space where the target is located. A search with three agents is shown below in Figure 8.

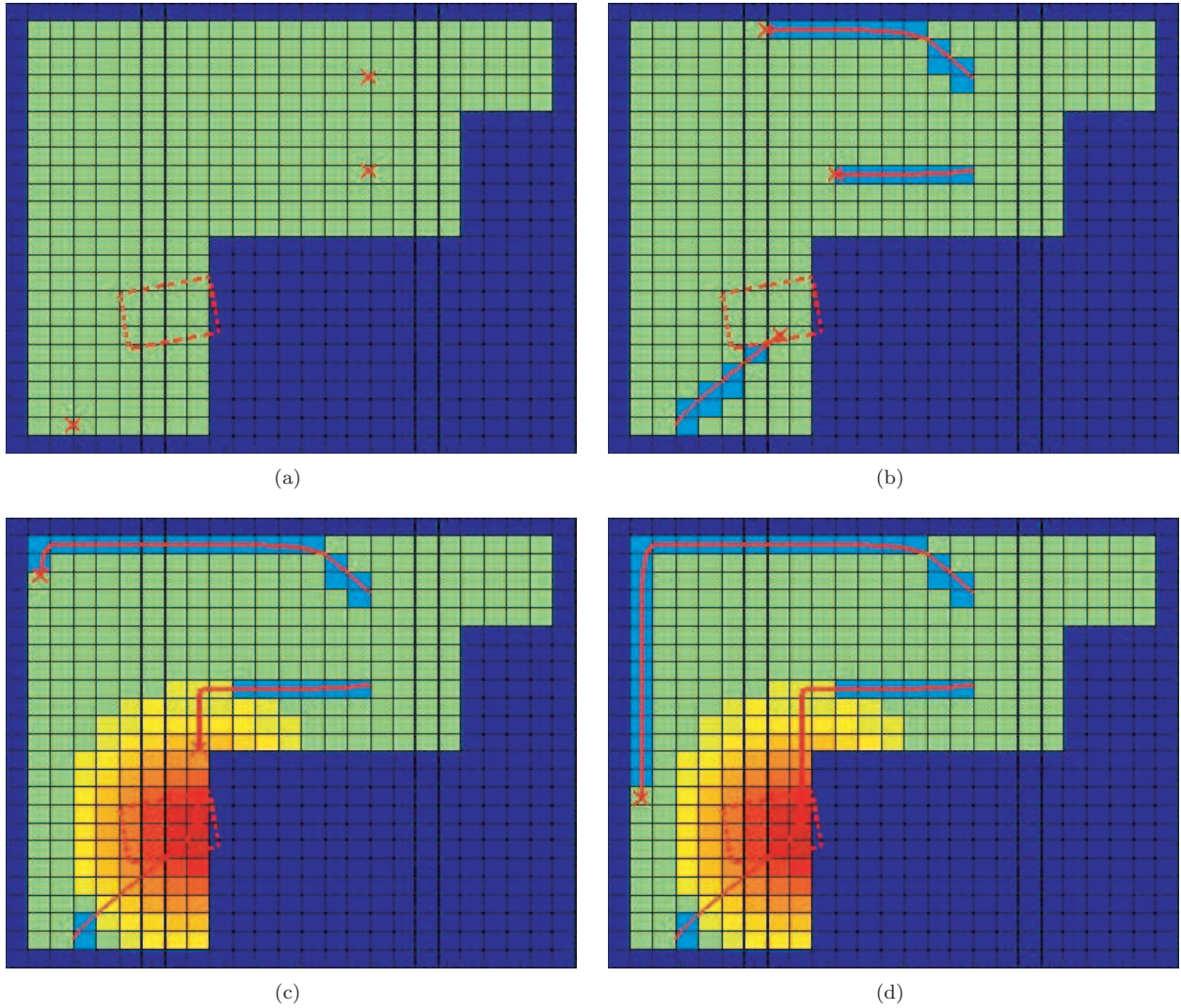


Figure 8. Occupancy based map search with three agents.

In this situation, the target location is shown as the dashed, red box and the agents are represented by red x's. Figure 8(a) shows the initial location of the agents relative to the target. In Figure 8(b), one of the agents is about to encounter an anomaly. In this case, the anomaly happens to be the target and therefore, the particle filter is able to identify and classify this anomaly as the target and the agent makes a positive ID at its current cell. This then updates the nearby occupancy map cells with increased scores. This causes the second agent to converge and investigate this location as shown in Figure 8(c). However, the third agent continues searching the other regions of the map as seen in Figure 8(d).

### V. Conclusion and Further Research

The main challenge in searching for a target in a noisy environment is identifying the anomalies that are encountered by the different agents. A particle filter with different motion and sensor models is able to handle the task of target identification for a heterogeneous team comprised of agents with different capabilities and sensors. Current research is directed at exploring the possible use of neural networks or support vector machines (also known as kernel machines) to process a trace of the sum of the particle weights (Figure 6)

and provide a single scalar measure of the confidence that the encountered anomaly is or is not the target.

This measure of confidence is then used to update the centralized occupancy based map. Using a gradient-type method of increasing the scores in cells near the identified anomaly allows nearby units to converge on the target but allows other agents to continue to search other possible areas. Future work in this area involves tailoring the update pattern for a positive identification so that agents pass over the target in non-conflicting trajectories and/or trajectories which maximize the amount of new information collected.

## VI. Acknowledgements

This work is sponsored in part by the Washington Technology Center (WTC) under grants F04-MC2 and F05-MC3. Juris Vagners at the University of Washington and Matt Wheeler and Brad Schrick at The Insitu Group also contributed to this work.

## References

- <sup>1</sup>Pongpunwattana, A. and Rysdyk, R., "Real-Time Planning for Multiple Autonomous Vehicles in Dynamic Uncertain Environments," *AIAA JACIC*, December 2004, pp. 580–604.
- <sup>2</sup>Rysdyk, R., Lum, C., and Vagners, J., "Autonomous Orbit Coordination for Two Unmanned Aerial Vehicles," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Autonomous Flight Systems Laboratory, 2005, Submitted to the 2005 GNC Conference.
- <sup>3</sup>"The Insitu Group," GeoRanger UAV Specifications, [http://www.insitugroup.com/prod\\_georanger.cfm](http://www.insitugroup.com/prod_georanger.cfm).
- <sup>4</sup>"Autonomous Flight Systems Laboratory," Laboratory Information, <http://www.aa.washington.edu/research/afsl/>.
- <sup>5</sup>"The Insitu Group," Company Information, <http://www.insitugroup.com/>.
- <sup>6</sup>Fox, D., Hightower, J., Liao, L., and Schulz, D., "Bayesian Filtering for Location Estimation," *IEEE Pervasive Computing*, July 2003, pp. 23–33.
- <sup>7</sup>Bourgault, F., Furukawa, T., and Durrant-Whyte, H., "Coordinated Decentralized Search for a Lost Target in a Bayesian World," *Proceedings of the 2003 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems*, Australian Centre for Field Robotics, Las Vegas, NV, October 2003.
- <sup>8</sup>Flint, M., Polycarpu, M., and Fernandez-Gaucherand, E., "Cooperative Control for Multiple Autonomous UAV's Searching for Targets," *Proceedings of the 41st IEEE Conference on Decision and Control*, Las Vegas, NV, December 2002.
- <sup>9</sup>Unmanned Dynamics, Hood River, OR, *AeroSim Aeronautical Simulation Blockset User's Guide Version 1.0*.
- <sup>10</sup>Thrun, S., Burgard, W., and Fox, D., *Probabilistic Robotics*, MIT Press, 2005.
- <sup>11</sup>Klein, D. J. and Klink, J. O., "Mobile Robot Localization," Tech. rep., University of Washington, Seattle, WA, 2005.
- <sup>12</sup>Pongpunwattana, A., *Real-Time Planning for Teams of Autonomous Vehicles in Dynamic Uncertain Environments*, Ph.D. thesis, University of Washington, Seattle, WA, June 2004.