UAV Path Planning in Search Operations

Chapter · January 2009

DOI: 10.5772/6479 · Source: InTech

2 authors:

Farzad Kamrani
KTH Royal Institute of Technology

Rassul Ayani
KTH Royal Institute of Technology

16

UAV Path Planning in Search Operations

Farzad Kamrani and Rassul Ayani
Royal Institute of Technology (KTH)
Sweden

1. Introduction

An Unmanned Aerial Vehicle (UAV) is a powered pilotless aircraft, which is controlled remotely or autonomously. UAVs are currently employed in many military roles and a number of civilian applications. Some 32 nations are developing or manufacturing more than 250 models of UAVs and 41 countries operate some 80 types of UAVs (U. S. Department of Defense, 2005). By all accounts utilization of UAVs in military and civilian application is expanding both in the short term and long term.

The two basic approaches to implementing unmanned flight, remote control and autonomy, rely predominantly on remote data communication and microprocessor technologies (U. S. Department of Defense, 2005). Advances in these technologies, which have grown exponentially since introduction, have dramatically improved the capabilities of the UAVs to address more complicated tasks. Increasing availability of low-cost computational power will stretch the boundary of what is possible to accomplish with less oversight of human operators, a feature generally called autonomy.

In many civil and military flight missions the aircrafts freedom of action and movement is restricted and the path is predefined. Given the path, the control task of the aircraft is to generate the trajectory, i.e. to determine required control manoeuvres to steer the aircraft from one point to another. However, in some flight missions the path is not predefined but should dynamically be determined during the mission, e.g. in military surveillance, search & rescue missions, fire detection, and disaster operations. In this type of scenarios the goal of the UAV is to find the precise location of a searched object in an area of responsibility. Usually some uncertain a priori information about the initial location of the object is available. Since during the search operation this information may be modified due to new reports from other sources or the UAV's observation, the path of the UAV can not be determined before starting the mission. In this chapter, we try to address this problem and introduce a framework for autonomous and dynamic UAV path planning in search operations.

The rest of this chapter is organized as follows: section 2 describes the problem in general and the instance of the problem that we solve, section 3 presents history and related work, section 4 provides an overall description of the proposed solution, section 5 and 6 explains Sequential Monte Carlo (SMC) methods and how we have applied it, in section 7 simulation of a test case scenario and the obtained results are presented, and section 8 summarizes the chapter.

2. Problem Definition

In surveillance or search and rescue missions an area of responsibility is assigned to a UAV with the task to find a target object. Usually the area is large and the detection time to find the object is the critical parameter that should be minimized. If no information about the target and the area of responsibility is available, then the only strategy is to exhaustively and uniformly search the area. However, in real life usually some information is available that justifies that the search effort is not evenly distributed over the entire area. An example of such situation is when some uncertain information about the former location of the target is available. This information may be combined with assumptions about the velocity and movement of the target to yield a time-dependent probability of the location of the target. Another situation is when sensor observations or report from other sources exclude some parts of the area. Furthermore, if the geographical map of the area of responsibility is available, one could use this information to concentrate search efforts on parts of the area, where the target is more likely to be found. Usually management of these fragments of information is performed by human operators, especially those with high experience in the field. The overall aim of the operator is to increase the utilization of the UAV resources by conducting the search operation in a manner that areas with higher probability of finding the object are prioritized and/or searched more thoroughly. When during the mission new information becomes available, it is required to repeat the procedure and modify the path if necessary. There are two major drawbacks with this approach. Firstly, since both the target and the sensor (UAV) are mobile, it is not always a trivial task to determine high probability areas and find the appropriate path. Analysis of this information may be beyond the capacity of a human brain. Secondly, due to the time-critical nature of these missions, it is not feasible to assign this task to an operator. Valuable time may be lost before information is processed by the human operator and it may be impossible to fulfil the time requirements, specially, where the information changes frequently or its volume is very high.

A more efficient approach is to automate the path planning process and integrate the reasoning about the locations of the target into the autonomous control system. In order to make this possible all available information has to be conveyed to the UAV and the autonomous control system should dynamically plan and modify the route.

In this work a simulation based method is introduced to address UAV path planning in search and surveillance missions, where some uncertain a priori information about the target and environment is available. Although the suggested framework is applicable in more general contexts, we have implemented and tested the method for a scenario in which a UAV searches for a mobile target moving on a known road network.

3. Related Work

Work on modern search theory began in the US Navy's Antisubmarine Warfare Operations Research Group (ASWORG) in 1942 (Morse, 1982; Stone, 1989). Bernard Osgood Koopman is credited with publishing the first scientific work on the subject in 1946, Search and Screening, which was classified for ten years before it was published (Stone, 1989). He defined many of the basic search concepts and provided the probabilistic model of the optimal search for a stationary target. However, developing algorithms for optimal search plan for moving targets started in the early 1970s and when computer technology became more available. The next step in developing search planners was to consider the dynamic nature of the search process. Computer Assisted Search Planning (CASP), developed for US Coast Guard in the 1970s by Richardson is a pioneer software system for dynamic planning of search for ships and people lost at sea (Richardson & Discenza, 1980; Stone, 1983). CASP employed Monte Carlo methods to obtain the target distribution using a multi-scenario approach. The scenarios were specified by choosing three scenario types and the required parameter values for each scenario. A grid of cells was used to build a probability map from the target distribution, where each cell had a detection probability associated with it. A search plan was developed based on the probability map. Feedbacks from the search were incorporated in the probability map for future search plans, if the first search effort did not succeed. The shortage of computer power and display technique did not allow CASP to be a truly dynamic tool operating in real-time aboard aircraft. Advances in computer technology provided the possibility of developing more feasible tools, Search and Localization Tactical decision aid (SALT) was a prototype air-antisubmarine search planner system for real-time use aboard aircraft (Stone, 1989).

The problem of searching for a lost target at sea by a single autonomous sensor platform (UAV) is discussed by (Bourgault et al., 2003a). In this paper the target may be static or mobile but not evading. The paper presents a Bayesian approach to the problem and the feasibility of the method is investigated using a high fidelity UAV simulator. Bayesian analysis is a way to recursively combine the motion model of the target and the sensor measurements to calculate the updated probability distribution of the target. Time is discretized in time steps of equal length and the distribution is calculated numerically. The search algorithm chooses a strategy that minimizes the expected time to find the target or alternatively maximizes the cumulative probability of finding the target given a restricted amount of time. The paper chooses one-step lookahead, i.e. the time horizon used for optimization is one time step. Because of this myopic planning, the UAV fails to detect the target if it is outside its sensor range. A decentralized Bayesian approach is suggested to solve the same problem by coordinating multiple autonomous sensor platforms (UAVs) in (Bourgault et al., 2003b; Bourgault et al., 2004). The coordinated solution is claimed to be more efficient, however, the simulations in these papers suffer from the short time horizon as well, i.e. one-step lookahead.

The problem of path planning of UAVs in search and surveillance missions (sensor platform steering) can be considered as a sensor resource management problem and is investigated by the Information fusion community as well. Sensor management is formally described as the process of coordinating the usage of a set of sensors or measurement devices in a dynamic, uncertain environment to improve the performance of data fusion and ultimately that of perception. A brief but highly insightful review of the multi-sensor management is presented by (Xiong & Svensson, 2003). The idea of simulating the target's future movements and choosing sensor control parameters to maximize a utility function is described in (Ahlberg et al., 2004). Given a situation X_0, all possible future situations X that are consistent with the positions in X_0 at time $t = 0$ are generated. For each of these X's, the utility of each sensor control scheme S is calculated by simulating observations of X using scheme S. The S whose average over all X is "best" is then chosen. However, to overcome the computation complexity, the set of possible sensor schemes is kept relatively small. Simulation-based planning for allocation of sensor resources is also discussed by (Svenson & Mårtenson, 2006). For each considered sensor allocation, the possible future paths of the target are partitioned into equivalence classes. Two futures are considered equivalent with respect to a given sensor allocation if they would give rise to the same set of observations. This partitioning to equivalence classes decreases the computational complexity of the problem and speeds up the calculation process.

4. Our Approach

The approach employed in this work to address the path planning problem is simulation-based. In short, this approach can be described as a method that uses simulation to approximate the future state of the target and tests alternative paths against the estimated future by running what-if simulations. These what-if simulations are conducted continuously during the mission (on-line). Utilizing information, even when it is incomplete or uncertain, is essential in constructing efficient search strategies and a system that uses all pieces of information in general performs better compared to systems not considering this information. In order to utilize this information, modeling and simulation techniques are used, which have shown to be a feasible tool handling complex and "difficult-to-analyze" systems.

The on-line simulation method for path planning in a search mission can be described as the following. The mission length is divided by a sequence of *time check points*, $\{t_0, t_1, ...\}$ where t_0 is the start time of the mission. At time t_0 the UAV chooses a default (random) path. At each time check point $t_k \in \{t_0, t_1, ..., t_n\}$, a set of simulations are started. In each simulation the state of the target for time $t \geq t_{k+1}$ and the effect of choosing an alternative UAV path for time $t \geq t_{k+1}$ are estimated. These simulations are completed during the time period $[t_k, t_{k+1}]$ and the results of these simulations are compared to choose the most appropriate path. At time t_{k+1} the chosen path is applied and a new set of simulations are started. Observations and other received information continuously modify the estimation of the target, but this updated model is employed when the UAV reaches the next time check point. That is observations obtained in time period $[t_k, t_{k+1}]$ affect simulations conducted in period $[t_{k+1}, t_{k+2}]$ which determine the path of the UAV after time t_{k+2}.

Apart from difficulties in constructing an on-line simulation system in general, some other problems should be addressed before this method can be employed in UAV path planning. Given the state of a system, the aim of on-line simulations is to predict the future state of the system and choose a course of action that is most beneficial for the system. In a surveillance mission, the state of the system (target) is not available. Indeed, the objective of the on-line simulation in this case is to optimize the process of acquiring information. The sensor data, before the target is detected, consists mostly of "negative" information i.e. lack of sensor measurement where it was (with some probability) expected (Koch, 2004). This information should be utilized to modify our estimation of the target's location. The process of drawing conclusions from sensor data is a problem studied by the information fusion community. One powerful estimation technique used in information fusion is Sequential Monte Carlo (SMC) methods also known as Particle Filtering which is based on point mass (or particle) representation of probability densities (Arulampalam et al., 2002).

In tracking applications, SMC is an on-line simulation process, which runs in parallel with the data collection process. In on-line UAV path planning we use SMC methods to estimate the current state of the target. This estimation (particle set) which is our only picture of the reality and is updated continuously is employed in "what-if" simulations to determine how the UAV should move to collect new data as effectively as possible. This path planning algorithm consists of two parts. The first part is a main loop running in real-time in which information is collected and our picture of the state of the system is updated. The other part is a set of "what-if" simulations that are initiated and executed periodically and after reaching time check points. These simulations run faster than real-time and are executed concurrently. Comparing these simulation outputs determines the most appropriate course of action. In the two next sections we describe SMC methods briefly and explain how it is applied in the path planning framework.

5. Sequential Monte Carlo Methods

In order to analyze a dynamic system using a sequence of noisy measurements, at least two models are required: First, a transition model which describes how the system changes over time and second, a sensor model which relates the noisy measurements to the state (Arulampalam et al., 2002). Usually these models are available in probabilistic form and since measurements are assumed to be available at discrete times, a discrete-time approach is convenient. In this approach the transition model, $p(x_k | x_{k-1})$, gives the conditional probability of the state x_k given x_{k-1}. The sensor model, $p(z_k | x_k)$, gives the conditional probability of observation z_k, given the state x_k. We are usually interested in the conditional state of the system, given the sequence of observations $z_{1:k} = \{z_1, z_2, ..., z_k\}$, i.e. $p(x_k | z_{1:k})$. In general, this conditional probability density function, may be obtained recursively in two stages, prediction and update. The prediction is calculated before the last observation z_k is available

$$p(x_k | z_{1:k-1}) = \int p(x_k | x_{k-1}, z_{1:k-1}) p(x_{k-1} | z_{1:k-1}) dx_{k-1} = \int p(x_k | x_{k-1}) p(x_{k-1} | z_{1:k-1}) dx_{k-1}. \tag{1}$$

The first equality follows from $p(x_k) = \int p(x_k | x_{k-1}) p(x_{k-1}) dx_{k-1}$ and the second equality is a result of the fact that the process is Markovian, i.e. given the current state, old observations have no effect on the future state (Arulampalam et al., 2002).
In the update stage the conditional probability $p(x_k | z_{1:k})$ is calculated using the prediction result when the latest observation z_k becomes available via Bayes' rule:

$$p(x_k | z_{1:k}) = \frac{p(z_k | x_k) p(x_k | z_{1:k-1})}{p(z_k | z_{1:k-1})} \tag{2}$$

where the denominator is calculated using

$$p(z_k | z_{1:k-1}) = \int p(z_k | x_k) p(x_x | z_{1:k-1}) dx_k. \tag{3}$$

If the transition model and the sensor model are linear and the process noise has a Gaussian distribution, which is a rather restrictive constraint, these calculations can be performed analytically by using Kalman Filter, otherwise some approximate method such as Particle Filtering (SMC methods) should be used (Arulampalam et al., 2002).

SMC methods are a set of simulation-based methods, which have been shown to be an appropriate tool for estimating the state of a non-linear system using a sequence of noisy measurements. Intuitively, SMC methods are simulations of how the state changes according to the transition model, and filtering the result using the sensor model. Since the system changes over time, this process is repeated in parallel with the real system when new data is received. Even if new observations are not available the prediction stage still can be used to predict the future state of the system. The procedure would be the same with the exception that since future measurements are not known yet, the update stage is not performed.

In SMC methods the probability density function of the state of the system in each time-step k is represented as a set of n points x_k^i in the state-space and corresponding weights w_k^i , i.e. $p_k^i = \{(x_k^i, w_k^i)\}_{i=0}^n$ where p_k^i is particle number *i* in time $t = k$.

The simulation begins with sampling S_0 , a set of n particles, from the a prior distribution $p(x_0)$, such that

$$S_0 = \{(x_0^i, w_0^i)\}_{i=1}^n, w_0^i = \frac{1}{n} \tag{4}$$

and number of particles in each interval [a, b] is in proportion to $\int p(x_0)dx_0$. At each iteration, particles in the set S_{k-1} are updated using the transition model, i.e. by sampling from

$$p(x_k^i \mid x_{k-1}^i) \tag{5}$$

and when observations arrive the weights are recalculated using

$$w_k^i \propto w_{k-1}^i p(z_k \mid x_k^i). \tag{6}$$

Particles are resampled periodically considering their weights, i.e. they will be sampled with replacement in proportion to their weights and weights are set to $w_k^i = 1/n$. This step is necessary to replicate particles with large weights and eliminate particles with low weights and avoid degeneracy of the algorithm (Arulampalam et al., 2002).

6. Applying SMC methods to UAV Search

One natural application area of SMC methods is target tracking and surveillance. The transition model is then derived from properties of the target, terrain characteristics and other forehand information available about the target. The sensor model depends on the characteristics of the sensors and the signature of the target. Many examples of applying

SMC methods in surveillance are provided in (Doucet et al., 2001; Ristic et al., 2004). Examples of applying the methods in terrain-aided tracking are found in (Ristic et al., 2004).
To demonstrate how the SMC methods work in practice, we present briefly how we have implemented them in the suggested framework. Here we assume that a single target is moving on a known road network. Some uncertain information about the initial location of the target, an approximation of its velocity and some assumption about its goal are available. The SMC methods include the following four stages: sampling, prediction, update, and resampling.

6.1 Sampling

The simulation starts with sampling $S_0 = \{(x_0^i, w_0^i)\}_{i=1}^N$ particles randomly from the a priori information $p(x_0)$, such that the number of particles on each (small) road segment is proportional to the probability of existence of the target on that road segment. Each particle is assigned a velocity randomly sampled from the distribution of the target's velocity. The weights of all particles are set equally to $1/N$.

6.2 Prediction

At iteration k, particles in the set S_{k-1} are propagated forward, that is the new state of the particles are calculated using their current location, velocity and a process noise based on the transition model, $p(x_k \mid x_{k-1})$. Since the motions of the particles (vehicle) are constrained by a known road network, their state can be specified by the vector $x_k = [r_k, d_k, v_k]$, where r_k is the current road segment, d_k is the distance the particle has moved on road r_k and v_k is the instantaneous speed of the particle.

6.3 Update

As described in the previous section, the sensor model is generally described by the probabilistic model $p(z_k \mid x_k)$, where x_k is the state of the system, and z_k is the observation at time $t = k$. The dimensions of the state z_k are usually, but not necessarily, less than the dimensions of x_k, since the system is not completely observable. We choose to distinguish between the part of the system-state which is not under our control, i.e. the state of the target x_k and the state of the UAV (sensor), y_k. Hence the probabilistic sensor model would be $p(z_k \mid x_k, y_k)$. We assume that the video interpretation task is solved by some means, i.e. we have a sensor that analyzes the incoming video at a constant rate and alarm with some probability if any target is observed. That is $z_k \in [ALARM, \neg ALARM]$.
Inspired by (Lichtenauer et al., 2004) we suggest the following model for sensor observations at a standard height.

$$p(ALARM \mid x_k, y_k) = \begin{cases} p_d & d \le \delta_{in} \\ p_d - \dfrac{(p_d - p_f)(d - \delta_{in})}{\delta_{out} - \delta_{in}} & \delta_{in} \le d \le \delta_{out} \\ p_f & d \ge \delta_{out} \end{cases} \quad (7)$$

In this model given the position of the target x_k and the position of the sensor y_k, the probability that the sensor indicates an ALARM, is calculated using four sensor constants. These characteristics are: detection probability (p_d), false alarm probability (p_f), inner detection range (δ_{in}), and outer detection limit (δ_{out}), see Figure 1.

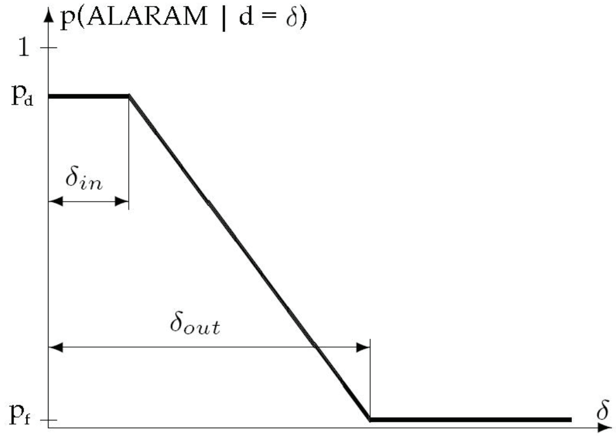

Figure 1. A graphic representation of the sensor model

After the propagation the weights of the particles are modified depending on the sensor model and current sensor observation. A sensor signal in a point increases the importance (weights) of the particles near that point. On the contrary, lack of sensor signals decreases the weights of the particles which are near the sensor. For instance if we have a perfect sensor and the UAV flies over a road segment and no sensor signal is supplied, the weights of all particles in that road segment are set to zero. After modifying the weights, they are normalized by dividing by the sum of the weights.

6.4 Resampling

A common problem with SMC methods is the degeneracy phenomenon, which refers to the fact that after many iterations, the number of particles with negligible weight increases and a large computational effort is devoted to updating particles, whose contribution to $p(x_k \mid z_k)$ is almost zero. One method to reduce the effect of degeneracy is to resample particles, i.e. to eliminate particles that have small weights and concentrate on particles with large weights. The resampling algorithm generates a new set of particles by sampling (with replacement) N times from the cumulative distribution function of weights of particles. The weights of these new particles are assigned the value 1/N.

7. Implementation and Test of a Scenario

The suggested path planning algorithm consists of two loops, the main control loop that includes the UAV and interacts with it at each time check point to modify the path of the

UAV to the *best known path*, and a simulation loop that estimates a picture of the reality, i.e. the probability density function of position of the target. At each time check point this picture of the reality is used in a series of what-if simulations with different possible paths of the UAV. The path that decreases the amount of uncertainty about the future is considered to be a "better" path.

Information entropy (Mackay, 2005), which is a measure of uncertainty associated with a discrete random variable $X \in \{x_1, x_2, ..., x_n\}$ is defined as $H(X) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i)$.

$H(X)$ takes only non-negative values, where $H(X) = 0$ indicates no uncertainty and larger values correspond to higher uncertainty. We suggest the expectation of the information entropy, $E[H(X)]$, as an objective function for comparing candidate UAV paths. In each step the path that decreases the expectation of the information entropy is chosen. SMC methods, which estimate the location of the target with a set of particles, provide an appropriate mechanism to estimate the expectation of the information entropy. Consider the UAV, being at point A in Figure 2(a), is facing the decision of whether to choose path ABC or ADE. The current estimate of the location of the target is shown by particles on these road segments.

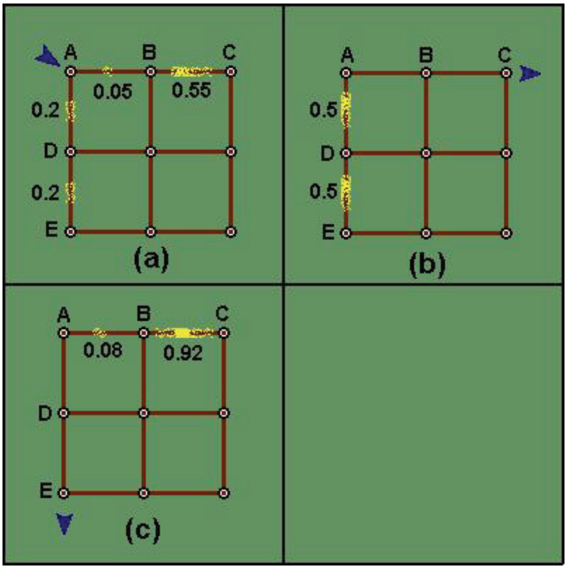

Figure 2. The impact of choosing different UAV paths on distribution of particles

Despite the fact that the total probability of finding the target on road ABC is 0.6, the most favourable path for the UAV is to choose ADE. Comparing Figures 2(b) and 2(c), shows that expectation of the information entropy by choosing path ABC is much more than choosing path ADE. See equations 8 and 9.

$$E[H(path = ABC)] = 0.p_{ABC} + (1 - p_{ABC})(-p_{AD} \log_2 p_{AD} - p_{DE} \log_2 p_{DE}) = \\ 0.4(-0.5 \log_2 0.5 - 0.5 \log_2 0.5) = 0.4 \tag{8}$$

$$E[H(path = ADE)] = 0.p_{ADE} + (1 - p_{ADE})(-p_{AB} \log_2 p_{AB} - p_{BC} \log_2 p_{BC}) = 0.6(-0.08 \log_2 0.08 - 0.92 \log_2 0.92) = 0.24 \tag{9}$$

To evaluate the performance of the suggested method, a test scenario is designed and simulations are performed using a special purpose simulation tool, called S2-simulator, introduced in (Kamrani et al., 2006). The tool contains a "real-world" simulator including a two dimensional terrain, a target object, and a UAV that can employ different search methods, one of which is on-line simulation method as described here. The on-line simulation method employs the simulation of this "real-world" to search for the target. Clearly, the information about the location of the target in the "real-world" is not available for the UAV. However, for simplicity we assume that some part of UAV's perception from reality is exact, e.g. the map of the terrain is accurate. Terrain is modelled as a two-dimensional landscape and includes two basic elements, nodes and road segments used to build a road network.

The geography of the test scenario, as shown in Figure 3, consists of a regular road network of perpendicular crossroads. Each road segment is 15 km long and the 60 road segments make an area of responsibility that covers a square of size 75 km by 75 km. For convenience a 2D coordinate system that has the origin located at the upper left-most node, with x values increasing to the right and y values increasing downwards is introduced.

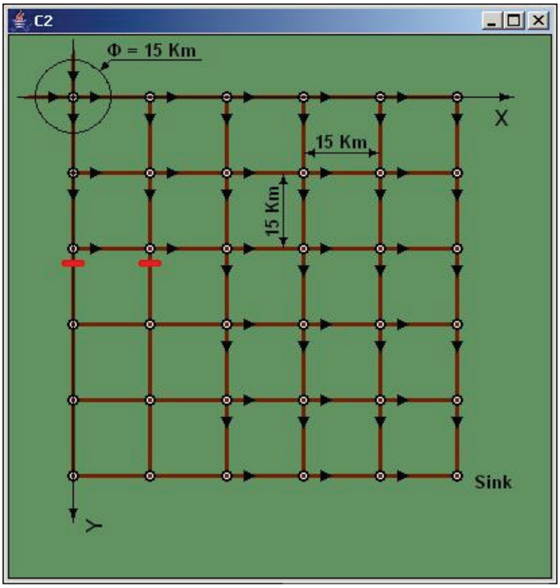

Figure 3. The target starts at the upper left-most node moving towards the sink

The target is initially located at the upper left-most node at origin. At this node and all other nodes the target moves either to east or toward south with equal probability, if any of these options are available. Hence, after passing 10 nodes and traversing 150 km the target reaches the lower right-most node and stops there. Considering these directions, the road network can be modelled as a directed acyclic graph with a source at origin and the sink located at the node in (75 km, 75 km). During the mission the road network may change as a result of the presence of friendly units, e.g. tanks may block certain roads. For example, in the scenario in Figure 3, two vertical road segments are blocked, which is shown by red bars in the figure. As seen in the figure, blocking a road segment does not affect only the blocked road segment but may change the distribution of particles in other areas. The target has a predefined path, which is unknown to the UAV. If a road in the path of the target is blocked the target chooses an alternative path. If directions to east and south are both blocked, the target stops. The average velocity of the target is 20 m/s with a velocity noise, thus it reaches its goal after approximately 7500 seconds. This velocity is unknown to the UAV.

The UAV starts its mission from a point in the III quadrant on the line y = x with a distance u_0 from the origin. Velocity of the UAV is 100 m/s. A large distance between the initial location of the UAV and the area of responsibility ensures that the target has a lead over the UAV. For example if u_0 = 180 km, the UAV reaches the area of responsibility after 1/2 hour.

The information available to the UAV system consists of the approximate initial location and velocity of the target, i.e. it is known that the target starts from a point uniformly distributed on the roads passing origin having a maximum distance of 7.5 km and has an average velocity between 15 and 25 m/s. It is as well known, that the target chooses one of the unblocked outgoing roads (if more than one) to south or east and there is no reason to believe that the target prefers one of these outgoing roads.

In the on-line simulation every 50 seconds (time check points), a new set of 60 alternative paths are compared by simulation. The length of these simulations is 1200 seconds. These simulations are run with the maximum possible speed, which depends on the computational power of the host computer. Simulations of the real world are run by time factor 5, i.e. 5 simulated seconds are equal to 1 (wall-clock) second. All simulations are run 10 times and the presented values are the average of these results.

Figure 4 illustrates the detection time of the target as a function of the distance of the UAV to the origin. Road network and the target's paths are also depicted in the same figure for 3 different cases. Blocked road segments are marked by red bars. Blocking of road segments occurs always during the mission, but before the target reaches the blocking points.

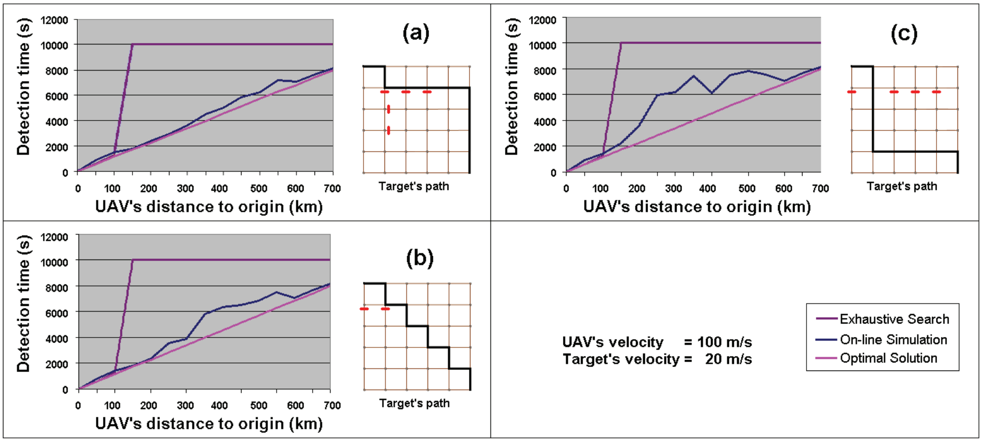

Figure 4. Detection time as a function of UAV's distance to the origin in online simulation compared with exhaustive search and the optimal solution

The values on the horizontal axis show the distance u_0 of the UAV with the origin, beginning from 0 to a maximum of 700 km. The vertical axis shows the average time for detection of the target. If the UAV fails to detect the target in 10000 seconds, the search is stopped. In each case the detection time for an exhaustive search method is also shown. In the exhaustive search method the UAV searches the entire road network indiscriminately starting from the road segments closest to origin and moving to the segments that are further from the origin.

Furthermore, a lower bound for the optimal value of "detecting" the target is also drawn in Figure 4. By optimal, we mean the time the UAV needs to meet the target if the path and the velocity of the target are completely known to the UAV. As expected, the detection time increases by increasing the distance of the UAV to the origin. However, the detection times for on-line simulation method remain significantly under 10000 seconds for all distances less than 700Km, while the exhaustive search method fails to detect the target when the UAV starts from a distance larger than 150 km. Moreover, comparing the result of the on-line simulation with the "optimal" solution in Figures 4-a, 4-b and 4-c, indicates that in some intervals the detection time approaches the optimal value, i.e. the time needed to reach the target if the UAV has all information about the target and the path.

8. Summary and Conclusion

In this chapter, we investigated the problem of autonomous UAV path planning in search or surveillance mission, when some a priori information about the target and the environment is available. A search operation that utilizes the available uncertain information about the initial location of the target, terrain data, and reasonable assumptions about the target movement can in average perform better than a uniform search that does not incorporate this information. We introduced a simulation-based framework for utilizing uncertain information in path planning. Search operations are generally dynamic and should be modified during the mission due to sensor observations, changes in the environment, and reports from other sources, hence an on-line simulation method was suggested. This method fuses continuously all available information using Sequential Monte Carlo methods to yield an updated picture of the probability density of the target's location. This estimation is used periodically to run a set of what-if simulations to determine which UAV path is most promising. From a set of different UAV paths the one that decreases the uncertainty about the location of the target is preferable. Hence, the expectation of information entropy is used as a measure for comparing different courses of action of the UAV.

The suggested framework was applied to a test case scenario involving a single UAV searching for a single target moving on a known road network. The performance of the method was tested by simulation, which indicated that the on-line path planning has generally a high performance. The result obtained by the on-line simulation method was compared with an exhaustive search, where the UAV searched the entire road network indiscriminately. The on-line simulation method showed significantly higher performance and detected the target in a considerably shorter time. Furthermore, the performance of the method was compared with the detection time when the UAV had the exact information about the initial location of the target, its velocity, and its path (minimum detection time). Comparison with this value indicated that the on-line simulation method in many cases achieved a "near" optimal performance in the studied scenario.

9. References

Ahlberg, S. & Hörling, P. & Jöred, K. & Mårtenson, C. & Neider, G. & Schubert, J. & Sidenbladh, H. & Svenson, P. & Svensson, P. & Undén, K. & Walter, J. (2004). The IFD03 information fusion demonstrator. *Proceedings of the Seventh International Conference on Information Fusion*, Vol. 2, pp. 936–943, June 2004

Arulampalam, S. & Maskell, S. & Gordon, N. & Clapp, T. (2002). A tutorial on Particle Filters for on-line non-linear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, Vol. 50, No. 2, pp. 174–188, February 2002

Bourgault, F. & Furukawa T. & Durrant-Whyte, H.F. (2003a). Optimal Search for a Lost Target in a Bayesian World. *Proceedings of the 4th International Conference on Field and Service Robotics (FSR'03)*, volume 24, pp. 209–222, Lake Yamanaka, Japan, July 2003

Bourgault, F. & Furukawa, T & Durrant-Whyte, H.F. (2003b). Coordinated Decentralized Search for a Lost Target in a Bayesian World. *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 1, pp. 48–53, Las Vegas, Nevada, October 2003

Bourgault, F. & Göktogan, A. & Furukawa, T. & Durrant-Whyte H.F. (2004). Coordinated Search for a Lost Target in a Bayesian World. *Advanced Robotics*, Vol. 18, No. 10, pp. 979–1000

Doucet, Arnaud & de Freitas, Nando & Gordon, Neil. (2001). *Sequential Monte Carlo Methods in Practice*. Springer Verlag, ISBN 0-387-95146-0

Kamrani, Farzad & Lozano, Marianela Garcia & Ayani, Rassul. (2006). Path planning for UAVs using symbiotic simulation. *Proceedings of the 20th annual European Simulation and Modelling Conference, ESM'2006*, pp. 215–238, Toulouse, France, October 2006

Koch, Wolfgang. (2004). On 'negative' information in tracking and sensor data fusion: Discussion of selected examples. *Proceedings of the Seventh International Conference on Information Fusion*, Vol. 1, pp. 91–98, June 2004

Lichtenauer, J. & Reinders, M. & Hendriks, E. (2004). Influence of the Observation Likelihood Function on Particle Filtering Performance in Tracking Applications. *Proceedings of the 6th International Conference on Automatic Face and Gesture Recognition*. pp. 767- 772, May 2004

Mackay, David J. C. (2005). *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, pp. 22-46

Morse, Philip M. (1982). In Memoriam: Bernard Osgood Koopman, 1900-1981. *Operations Research*, Vol. 30, No. 3, May - June 1982, pp. 417–427

Richardson, Henry R. & Discenza, Joseph H. (1980). The United States Coast Guard Computer-Assisted Search Planning System (CASP), *Naval Research Logistics Quarterly*, Vol. 27, No. 4, March - April 1980, pp. 659–680

Ristic, Branko & Arulampalam, Sanjeev & Gordon, Neil. (2004). *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House Series Publishers, ISBN 1-58053-631-x

Stone, Lawrence D. (1983). The Process of Search Planning: Current Approaches and Continuing Problems. *Operations Research*, Vol. 31, No. 2, March – April 1983, pp. 207–233

Stone, Lawrence D. (1989). What's Happened in Search Theory since the 1975 Lanchester Prize? *Operations Research,* Vol. 37, No. 3, May - June 1989, pp. 501–506

Svenson, P. & Mårtenson, C. (2006). SB-Plan: Simulation-based support for resource allocation and mission planning. *Proceedings of the Conference on Civil and Military Readiness (CIMI 2006)*, Enköping, Sweden, May 16–18, 2006

U. S. Department of Defense. (2005). *Unmanned Aircraft Systems (UAS) Roadmap, 2005-2030*, August 2005, pp. 47-52

Xiong, N. & Svensson, P. (2003). Multi-sensor management for information fusion: issues and approaches. *Information Fusion*, Vol. 3, No. 2, pp. 163–186