In Search of a Lost Target via Monte Carlo Simulation and Automated Planning

Sara Bernardini and **Maria Fox** and **Derek Long**
Department of Informatics
King's College London
London, UK, WC2R 2LS
firstname.lastname@kcl.ac.uk

Abstract

Devising techniques for underpinning the behaviour of autonomous vehicles in search-and-tracking missions is particularly challenging as these vehicles operate in uncertain and unpredictable environments where they must cope with little stability and tight deadlines in spite of their restricted resources. State-of-the-art techniques typically use probabilistic algorithms that suffer a high computational cost in complex real-world scenarios. To overcome these limitations, we propose a *hybrid* approach that combines the probabilistic reasoning based on the target motion model offered by Monte Carlo simulation with long-term strategic capabilities provided by automated task planning. We demonstrate our approach by using a fixed-wing UAV deployed in simulation. Our experimental results show that our unique way of integrating probabilistic and deterministic reasoning pays off when we tackle realistic missions.

1 Introduction

In Search-and-Tracking (SaT) missions, an observer searches for a mobile target and tracks it once it has found it. Examples of SaT operations are a UAV searching for life-rafts drifting with current, a police helicopter tracking a suspected criminal over a road network and a small drone escorting a worker who performs risky tasks in a factory. As SaT has important applications in a number of critical real-world operations such as search-and-rescue and surveillance missions, recently there has been increasing interest in equipping UAVs with autonomous SAT capabilities.

Autonomous SaT is a difficult problem for a number of reasons. Observers usually operate in unpredictable environments with little stability and rapidly changing information. They must decide what action to perform and how to coordinate with other observers almost instantaneously. They must be highly trained to react quickly, without spending too much time reasoning about alternative courses of action. At the same time observers have limited resources, e.g. fuel or battery, and need to be strategic in deciding what course to ride, looking ahead at their remaining lifespan and fitting their objectives within this time frame. Hence, SaT gives rise to many challenges: the management of uncertainty in an unpredictable environment, the handling of restricted resources, the right balance between reactivity and deliberation and the communication of requests and commitments between multiple heterogeneous observers, including human operators in mixed-initiative scenarios.

Although many techniques have been used to address these challenges separately, in our previous work (Bernardini et al. 2013; Bernardini, Fox, and Long 2014), we show that *automated task planning* is ideally suited to deal with all these requirements at the same time. Planners offer a route to crafting effective strategies for the observers to achieve their mission goals in the face of all relevant constraints, such as restricted resources, tight deadlines and uncertainty. The use of task planning for SaT missions has received little attention so far, while probabilistic approaches based on Recursive Bayesian Estimation (RBE) have been explored in more depth (Bourgault, Furukawa, and Durrant-Whyte 2003; Furukawa et al. 2006). However, they rely on restrictive simplifying assumptions such as the search area being small, the temporal horizon being short and the target's motion model being simple. Therefore, they usually fail in the face of all the constraints that characterise real-world SaT operations.

In (Bernardini et al. 2013), we introduce a plan-based approach to SaT for large-scale and long-term missions. We formulate the search phase of a SaT mission as a *deterministic* planning problem and use an off-the-shelf automated planning tool to solve it and generate robust strategies for the observer. Since plans are always produced under the assumption that the target is lost and its position is unknown, we can largely ignore the probabilistic aspects of the search problem and get away with a deterministic formulation of it. Our technique leads to good policies when the target's behaviour is predictable, but incurs inaccuracies when the target acts in a more sophisticated way, as we neglect important information about the physical motion of the target.

In this paper, we combine our previous plan-based approach to SaT with *Monte Carlo* (MC) methods. We take our approach one step further by integrating it with probabilistic reasoning based on the target's motion model and on the environment's structure. In our novel *hybrid* technique, we apply MC simulation (MCS) to estimate the target probable trajectories and construct a fine-grained Probability Distribution (PD) map for the target location, while using planning to reason about this map and create long-term strategic plans for the observer that maximise the probability of rediscovering the target. The MCS works on the basis of historical information about the target, its physical model of movement and topological information about the area of operations. The experimental results presented in Section 7 show that our hybrid method outperforms other existing techniques as well as our previous plan-based approach.

2 Search-and-Tracking

A SaT mission aims to follow the target to its destination and proceeds in two phases, which constantly interleave: (i) *Tracking*: the observer flies over the target, observing its progress; and (ii) *Search*: the observer loses the target and flies a series of manoeuvres to rediscover it. Once the target is spotted, the observer switches back to tracking.

We are interested in land SaT operations with a single observer and a single target. In our work, we remove the typical assumptions behind state-of-the-art probabilistic approaches to SaT (Furukawa et al. 2006; Hsu, Sun, and Rong 2008) and consider scenarios with the following features, which characterise the majority of realistic missions: (i) the target moves according to its own intentions; (ii) the target moves across a large geographical areas; and (iii) the target needs to be tracked over a long period of time. We assume that the target needs to reach a specific destination and chooses an efficient path to do so, without trying to evade the observer. This is a plausible assumption as the target might be cooperating with the observer or simply be unaware of its presence, but we recognise that the extension to consider evasive actions on the target's part is important future work. The observer is equipped with imaging systems to scan the search area and observe the target. Observation is susceptible to error and interference from the features in the environment.

3 Plan-based Approach to SaT

During the tracking phase of a SaT mission, the observer follows the target, observing its progress. At first sight, tracking might appear to be a planning problem, in particular a temporally extended goal problem where the goal is to keep the target in view as long as possible. However, as we consider tracking in more depth, we see that it is in fact a reactive control problem, since the target's intentions are unknown and the observer can only respond to the target's movements moment by moment. It is when the target is out of view that we need to carefully plan a recovery strategy to relocate it. Our approach is therefore to track the target reactively while it is visible and to plan a recovery strategy every time it is lost by using an automated planning tool. The observer executes a simple static policy over the course of the mission: (i) plan to find the target; (ii) track the target once it has been found; and (iii) re-plan, if target is lost again.

We manage the tracking phase through a *reactive controller* equipped with sensing capabilities. If the speed of the observer and the target are comparable, the observer simply flies over the target. However, if the observer flies much faster than the target and cannot hover, the flight path of the observer is a circle of fixed radius centred on the target. The radius depends on the observer's capabilities: it cannot be greater than the imaging equipment's range, nor can it be shorter than the observer's turning radius at current speed. We assume that the observer flies in a mid-range circle between these extremes. As the target moves, the circle moves with it, so the observer's flight path describes a prolate cycloid over the ground.

When the observer fails to follow the target, it must attempt to rediscover it. For a short period after losing the target, the observer simply tracks its predicted location, since the target cannot move fast enough to significantly deviate from this prediction. However, after a longer period, it is necessary to make a more systematic effort to rediscover the target by directing the search into specific places. This is when planning comes into play. We formulate the search phase as a *planning task* consisting of deciding where exactly to search for the target and what manoeuvres to use. The goal is to maximise the likelihood of finding the target while favouring manoeuvres that minimise the use of the observer's consumable resources. We use *MC simulation* to aid the planning process in making decisions by suggesting probable trajectories that the target might follow during its course of action.

4 Search Operations

In line with SaT and SaR (Search-and-Rescue) international standard guidelines (IMO 2013; NATSAR 2011; CSAR 2000), we employ the following procedure to manage the search phase of a SaT mission:

1. Determining the optimal area where the search effort should be deployed, which is an area where the target is most likely to be found;
2. Dividing this area into appropriate sub-areas for assignment to individual *search patterns*, which are sets of manoeuvres for surveying specified regions;
3. Selecting specific search patterns and their orientations to optimally cover each sub-area;
4. Determining a *sequence* in which the chosen patterns need to be executed; and
5. Executing the chosen sequence of patterns, switching back to tracking if the target is rediscovered.

Steps 1 and 2 depend on information regarding the specific mission. In real-world SaT operations, these steps are performed based on many biasing factors: the target last known position (LKP), its intentions if predictable, its size and motion characteristics, possible hazards, results of previous searches, the terrain characteristics, the road network's structure and the weather conditions. These features are used to make predictions on where the target might be over time and to construct a PD for the target location. In short, the outcome of Steps 1 and 2 consists of:

(i) a confined search area, usually a circular sector that is centred on the LKP of the target;

(ii) a PD for the target position defined in this sector and constructed considering the above-mentioned factors;

(iii) a number of points within the sector that present the highest probability of rediscovering the target and on which the search patterns should be deployed.

To calculate these three pieces of information, we perform MCS based on the target motion model as explained in detail in Section 5. In our previous work (Bernardini et al. 2013), we construct the target PD manually by considering the features of the road network and the terrain in the search area. This calculation is very efficient, but our experiments show that the plan-based approach suffers from the use of this coarse-grained distribution. Through MCS, on the other hand, we construct a much more precise PD map for the target location because the simulation is based on the target physical model of motion and incorporates information that reflects the environment structure as well as the target's intentions and physical characteristics (e.g. its minimum and maximum speed on roads). Our MC-based approach is very general as any new assumption regarding the target motion can be easily integrated in the simulation with the resulting PD expressing such additional information.

Step 3 is concerned with choosing a particular search pattern to cover each sub-area within the optimal area of operations. Again, we adhere to standard procedures (IMO 2013; NATSAR 2011; CSAR 2000) and use the following search patterns: (i) *Lawnmower Search* (LS), which consists in flying along straight lines with 180° turns at the end. We use lawnmowers, which can be of any dimension, if the search area is large and level, only the approximate location of the target is known and uniform coverage is desired. (ii) *Spiral Search* (SS), which we use if the search area is small and the position of the target is known within close limits; and (iii) *Contour Search* (CS), used to patrol obstacles, always assumed to be polygonal in our application.

In Step 4, a subset of the candidate search patterns generated in Step 3 needs to be selected and sequenced for execution (Step 5). While this task is usually managed manually in real missions, with the consequence of being a time-consuming and error-prone process, we formulate the problem of selecting and sequencing search patterns as a *planning problem* and use a high-performing planner to solve it. Our plan-based approach to select and sequence search patterns is explained in detail in Section 6.

5 Monte Carlo Simulation

In SaT, the planner's role is to select a set of search patterns and sequence them over time. To operate effectively, we need to provide the planner with an initial pool of *candidate search patterns* from which to choose those to execute. It is preferable to keep the cardinality of this set small so as to reduce the computational complexity. We perform MCS to identify points in the search area that present the highest probability of finding the target at different points in time and then create candidate patterns that have those points as their centres.

We assume that the target is located in Euclidean 2-space and that this space is characterised by a *road network* (RN), where each road is a sequence of connected line segments. The target is in uniform motion on such segments. We assign each road a PD over the speed that a moving object can assume while proceeding on the road within the range of a reasonable minimum speed of 20 mph and a maximum speed of 70 mph (these values are modifiable parameters).

We take a circular sector centred on the target's LKP as the optimal search area. This sector extends outwards with its symmetry axis aligned with the target's average bearing over the period it was observed. The radius of the sector is determined by considering both the target's travel speed and the time period over which the search is planned to be performed. We then superimpose a grid on this sector and, to represent the topology of the search area, build a graph $\mathcal{G} = \langle \mathrm{V}, \mathrm{E} \rangle$ based on the RN enclosed in the grid. We add a vertex for each cell in the grid and add edges only between vertices corresponding to adjacent cells. More specifically, we add an edge from a vertex v_i corresponding to a cell c_i to a vertex v_j corresponding to a cell c_j adjacent to c_i if and only if there is a segment in the RN that intersects both the cells c_i and c_j. Each edge is labelled with the travelling time that the target would take to go from the centre of cell c_i to the centre of cell c_j proceeding at maximum speed within the speed limits of the fastest road that connects c_i to c_j.

We assume that a set of target possible destinations is given as well as a PD over them. From the graph $\mathcal{G}$, we calculate the shortest path from the target LKP to each destination by using Dijkstra's single-source-shortest-path algorithm and store these paths in a table.

Based on these data structures, we perform MCS as follows. We generate a set of particles, where each particle represents a discrete guess of where the target might be located over time. We assign each particle a specific destination by randomly choosing it based on the PD of the target destinations. At any time point, each particle is structured as an x coordinate, a y coordinate, a speed and a heading direction. On each road traversed by the particle, the speed is randomly sampled from the PD of the target possible speeds for that type of road and the heading direction is given by the vector connecting the target LKP to its associated destination. Initially, all the particles are located at the target LKP. Then, for each particle, we simulate motion assuming that the particle proceeds towards its destination following the shortest path to it and going at a constant speed on each road segment. The underlying observation that motivates this simulation of motion is that, most of the time, a moving object that is assumed to be heading to a certain destination is likely to prefer to move in the general direction of the destination by following the main roads.

Given the total mission time T, we establish a set of time check points $t_0, t_1, \ldots, t_n$, where t_0 is the start of the mission and each $t_i < (t_0 + \mathrm{T})$. We then simulate the motion of the particles up to each time check point t_i and store their arrival positions for each temporal slice. In such a way, we obtain an approximate representation of where the target might be at the different time check points or, in other words, a PD for the target position at these times.

For each time check point, we select the positions with the highest probabilities and generate a set of candidate search patterns centred on those positions. As explained in Section 4, the specific type of pattern to use depends on many factors relating to the features of the area that the pattern covers.

Via MCS, we construct a fine-grained PD map for the target location. The particles end up distributing over the main roads and clustering around the destinations. Thanks to MCS, we generate only candidates that bear a high probability of rediscovery, allowing the planner to restrict its reasoning to a limited set of promising patterns. Although the MCS just described is based on our specific assumptions, the method can be easily modified to fit different assumptions.

6 Search as Planning

Once a set of candidates has been identified via MCS, the challenge is then to decide which of them should be selected for execution and when they should be executed. We see the task of selection and sequencing of search patterns as a *planning problem*. We assign each pattern a time-dependent *reward*, i.e. a value corresponding to the expectation of finding the target in a search of the area that the pattern covers. We calculate this value based on the PD for the target location obtained through MCS. Based on the patterns' rewards, the planner can select a sequence of patterns that maximises the accumulated expectation of rediscovering the target.

This planning problem has some unusual and interesting features. Despite the inherent uncertainty in the situation, the problem is *deterministic*, since the uncertainty arises in the position of the target and, if the target is found, the plan ceases to be relevant. Therefore, the plan is constructed entirely under the assumption that the target remains undiscovered. Somewhat counter-intuitively, "plan-failure" corresponds to the situation in which the target is found.

6.1 Planning Domain

We use the language PDDL2.2 (Edelkamp and Hoffmann 2004) to model the search problem, making advantage of several sophisticated features of this language to express all the properties of our domains. The basic structure of the domain for the search problem is simple: there is a flight action that allows the observer to fly from one waypoint to another and there are search actions corresponding to the flight patterns, i.e. spirals, small and large lawnmowers and contour searches. These search actions have a similar form (see Table 1): they have an entry waypoint and an exit waypoint and the effect, other than to move the observer, is to increase the total reward (i.e. the accumulated expectation of finding the target) by a quantity that represents the specific reward associated with that particular pattern. The actions are durative and their duration is fixed in the problem instance to be the computed value for the execution of the corresponding search. The search patterns can only be executed when they are active, i.e. during a window of opportunity that coincides with the period in which the target could plausibly be in the area that the pattern covers. This window is calculated by considering the minimum and maximum reasonable speeds for the target and the distance from the target LKP.

6.2 Problem Specification

The initial state of a planning problem contains all the candidate patterns from which the planner chooses the ones to execute. They are the objects of the planning instance. Each candidate search pattern is assigned the following information: (i) a fixed duration, which is the computed value for the execution of the corresponding search; (ii) an opportunity window, which specifies when the pattern is active; (iii) entry and exit waypoints; and (iv) a *reward*.

```
(:durative-action doSpiral
  :parameters (?from ?to - waypoint
               ?p - spiral)
  :duration (=?duration (timeFor ?p))
  :condition
   (and (at start (beginAt ?from ?p))
   (at start (endAt ?to ?p))
   (at start (at ?from))
   (at end (active ?p)))
  :effect
   (and (at start (not (at ?from)))
   (at end (at ?to))
   (at end (increase(reward)(rewardOf ?p))))
 )
```

Table 1: The action `doSpiral` encoded in PDDL2.2.

The reward depends on both the probability that the target enters the area being searched and the probability of detection, i.e. the probability that the observer sees the target while passing over it. This probability is affected by several factors, such as the type of search pattern, the camera used, the direction in which the target has been travelling and the characteristics of the area traversed by the target.

As we deal with a moving target, the reward is a *time-dependent* function. No reward is assigned until the target has plausibly arrived to the area covered by the pattern and once the target is deemed likely to have left the area. Between these extremes, the reward is modelled as a step function that approximates a lifted Gaussian distribution. It increases when the pattern becomes active, it increases further when the target is considered most likely to be in the pattern and then it decreases until the end of the useful life of the pattern. The reward peaks at the point where the target is assessed to be in the centre of the search pattern. Based on the PD over the area of search obtained through the MCS, the peak reward is calculated as a proportion of the PD over the time period covered by the search pattern. We use a unimodal distribution because we assume that the target moves towards its destination using the most efficient path to reach it, never revisiting locations that it has already traversed. The variance is generated by uncertainty about the precise path and precise speed of the target.

To ensure that the planner does not exploit search patterns when there is no reward associated with them, the patterns are only made active during the period when the PD is positive, using *timed initial literals* (TILs) that are asserted and retracted at the appropriate times. Reward is therefore modelled as a series of n times, t_0 to t_n. At each t_i a TIL asserts the value of the reward function for the interval $[t_i, t_{i+1}]$, with reward being set to 0 in the initial state and reset to 0 by the TIL at t_n. A fragment of the problem specification for the UAV domain is presented in Table 2.

Along with the initial state, the problem specification contains a description of the goal. The problem has no goal, but the plan metric measures the value of the plan in terms of the accumulated expectation of finding the target. This is specified as reported in Table 3.

```
(= (rewardOf spiral1) 361)
(at 983  (= (rewardOf spiral1) 723))
(at 1129 (= (rewardOf spiral1) 361))
(= (timefor spiral1) 299)
(beginAt s1s spiral1)
(endAt   s1e spiral1)
(at 253  (active spiral1))
(at 1349 (not (active spiral1)))
(= (distance origin s1s) 56)
(= (distance s1e s1s) 36)
(= (distance s1e s2s) 8)
(= (distance s1e s3s) 39)
 ...
```

Table 2: Part of the initial specification for the UAV domain.

```
(:goal (and (>= (reward) 1)))
(:metric maximize (reward)))
```

Table 3: Goal specification for the UAV domain.

6.3 Planning Mechanism

We exploit the period in which the observer tracks the target predicted location to perform planning. We use an off-the-shelf planner called OPTIC (Benton, Coles, and Coles 2012) to build plans for the observer. It performs anytime, cost-improving search: it finds a first solution very quickly, but it then spends the additional time improving on this solution by adding further manoeuvres to the plan or by trying different collections of manoeuvres. The search uses a weighted-$A^\star$ scheme with steadily changing weights in a tiered fashion. The plans produced are monotonically improving, so the final plan is selected for execution. We use a time-bounded search limited to 10 seconds because we are in a time-critical situation (although this value is also a configurable parameter).

Table 4 shows an example of a plan generated by OPTIC for the UAV domain. The plan is dispatched via a simple controller, action by action. At the conclusion of the plan, the observer abandons the search.

Time	Action	Duration
0:	(fly origin s1s)	[39]
39:	(dospiral s1s s1e spiral1)	[299]
338:	(fly s1e s2s)	[1]
339:	(dospiral s2s s2e spiral2)	[299]
638:	(fly s2e s2s)	[36]
674:	(dospiral s2s s2e spiral2)	[299]
973:	(fly s2e s3s)	[73]
1046:	(dospiral s3s s3e spiral3)	[1099]

Table 4: A plan generated by OPTIC for the UAV. The columns are: execution time, action and action duration.

7 Experimental Results

We demonstrate the viability of our hybrid approach to SaT by implementing our technique in simulation, which features a fixed-wing UAV involved in a complete SaT mission.

The UAV Simulator. The UAV simulator was built in consultation with our industrial collaborators and is intended to provide an appropriately abstracted view of the problem. The main abstraction is that we assume the control problem for the UAV solved.

The simulated UAV is equipped with imaging systems that allow the target to be observed and that is susceptible to error. The detection probability on each observation cycle (which can be considered as a 'frame capture' by the imager) depends on: terrain, speed, discrepancy between anticipated and actual target positions and the imaging system mode. The imager has two modes: wide-angle, used to increase the area being scanned when the target is not currently observed at the cost of a lower probability of successfully observing the target, and narrow-angle, in which the viewing area is reduced, but the probability of detecting the target is higher. The effect of terrain is to reduce the probability of spotting the target in urban, forested and mountainous areas, while in rough or open rural areas the probability is higher.

The area of operations is a part of Scotland about 100 kilometres square. Terrain types were defined by hand, along with an approximate RN for the major roads and rural minor roads. The target follows a path acquired using Google Maps, using a selected (configurable) origin and destination. This information is also used to decide what speed is appropriate for the target, based on distance between waypoints in the route proposed by Google Maps and terrain type.

Simulation Steps. During the tracking phase of the SaT mission, the UAV simulator follows the target by spiralling over it. Once the target has been lost and during the period in which the UAV tracks the predicted location of the target at low confidence, the simulator performs the following steps:

1. It runs MCS, which produces a PD map for the target location at different time steps;
2. It generates a set of candidate search patterns based on the PD map and synthesises a planning task specification featuring these candidates (Figure 1(a));
3. It feeds the domain and the problem into the planner, while displaying the stages of the planning process; and
4. After 10 seconds, it dispatches the generated plan to the UAV and simulates its execution (Figure 1(b)).

As for the MCS, we follow the procedure presented in Section 5, but incorporate in it additional assumptions that are specific to our simulator. We assume that the RN is characterised by three types of roads: motorways (the fastest roads), A roads (the main routes between towns) and B roads (the smallest of the three). We establish that the target average speed is 62 mph on motorways, 55 mph on A roads and 25 mph on B roads, with maximum and minimum speeds set appropriately around these values. After experimenting with different granularities, we now adopt a grid square size of 500 meters. The graphs extracted from the RN have around 30.000 nodes and 18.000 edges, on average. We choose as the set of possible destinations the first 15 most populated cities in Scotland and assign them equal probability. We generate 1000 particles and, since the total mission time for our application is about one hour, we consider 17 time check points spaced 150 seconds apart. The simulator visualises the most probable cells in the map at the different time check points in different colours (Figure 1). As expected, the particles follow the main roads and cluster around cities.

The simulation tool offers various opportunities for interaction, including redirecting the target, repositioning the observer, speeding and slowing the simulation and modifying the parameters that govern spotting probabilities, flight dynamics and the target behaviour.

Results and Discussions. We conducted a series of experiments to assess the performance of our hybrid approach to SaT by using the UAV simulation. It is difficult to compare our approach with purely probabilistic methods, as the assumptions behind them are very different from ours. Instead, we assess our strategy against two comparable ones: a fixed policy, which is used as a baseline for evaluating the benefits of a plan-based approach, and a plan-based approach that does not use MCS.

Our industrial collaborators proposed the fixed policy, which is used in several real-world applications. When the UAV loses the target, it tracks its predicted location for 3 minutes. If it has not found the target, it executes a fixed sequence of patterns: first a spiral around the target LKP and then a large lawnmower pattern (20 km square).

Though broadly similar, the plan-based policy (Bernardini et al. 2013) differs from our strategy in one crucial way: the target PD is not generated via MCS, but it is constructed by hand based on the features in the environment, without taking into account the target motion model. In particular, the PD is based on the density of roads across the search area, which is measured by using a fine-mesh grid and counting the number of significant roads within each grid cell, the terrain type and the distance from the target LKP. The distribution decays linearly with distance from the origin and is weighted by values for terrain type and RN density. Plan-based SaT may find good policies for problems with predictable target behaviours, but it struggles with serious inaccuracies when the target acts in a more sophisticated way by neglecting its physical motion in the environment.

Both for our strategy and for the plan-based one, we use a configuration that tracks the predicted location of the target for the same period as the fixed policy, before planning and executing a search plan. We generated 15 routes and executed the simulation on each route 1000 times (the simulation has a non-deterministic spotting model and target behaviour), for each of the 3 strategies (a total of 45.000 runs). The simulation begins with the target undetected, but in the search arc of the observer. In a small number of runs the observer fails to detect the target in the very early stage. Our simulation does not use a search plan in this first stage, so failure at this point leads to an early abort. We discount these runs (less than 0.5%) in our analysis.

Figure 2, which shows the proportion of runs in which the target was tracked to its destination, demonstrates that our hybrid strategy is consistently better than the other strategies. The fixed policy has an overall success rate of around 42%, the plan-based policy without MCS has overall success rate of around 56%, while the plan-based policy with the MCS yields better than 72% success rate. Our hybrid approach very particularly well when the target drives towards destinations far away from the origin as the MCS allows us to make better predictions than the other strategies.

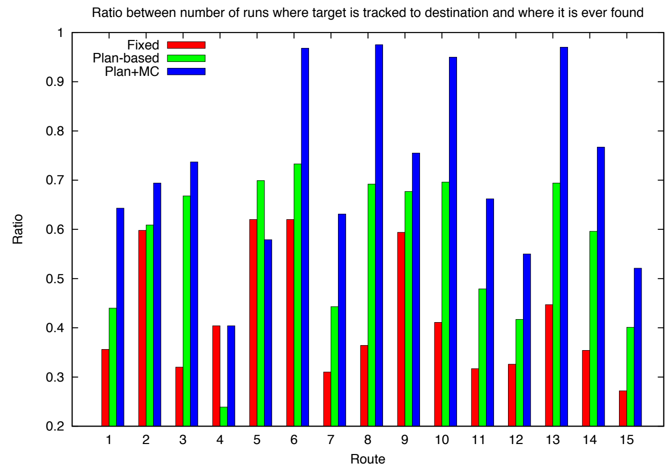

Figure 2: Proportion of successfully tracked targets.

Figure 3 shows the average time that the observer tracks the target plotted against journey duration for the three strategies. It can be seen that the hybrid technique produces the best performance, while the static policy is generally the weakest. Figure 4 shows the average time at which the target is lost for the last time, plotted against the duration of the journey. This figure shows that, when the simple plan-based policy is used, the target is lost, on average, at about the same time in the journey (after about half-an-hour) regardless of the duration of the journey. We conjecture that this is because these journeys all start in an urban area that it takes approximately 30 minutes to cross. During this period the target has a constrained heading due to the RN across town. On leaving, it often corrects its heading by a significant deviation and this is when the observer loses track of it. On the other hand, when the plan-based policy is used in combination with the MCS, the time at which the target is lost for the last time increases with the increase of the duration of the journey. This is because, on average, our strategy is capable of tracking the target for a longer time.

Our results clearly demonstrate the benefit of using planning in combination with MCS both with respect to fixed policies, currently being employed in real-world scenarios, and with respect to a plan-based approach in which the PD maps are not accurate enough to allow planning to express its full power.

8 Related Work

Real-world systems. The first practical issues relating to search for a lost target were posed by B. Koopman in the US Navy during World War II (Koopman 1946) and revolved around providing efficient methods of detecting submarines. Since then, there has been extensive work in theory of search (Stone 1975) and Coast Guards around the world

(a) Initial State (b) Plan

Figure 1: Screenshot 1(a) shows an initial state: circles are the spirals that the planner will consider. Colour intensities are proportional to probabilities, with more intense colour corresponding to higher probability. Screenshot 1(b) shows the search patterns that have been selected by the planner for execution.

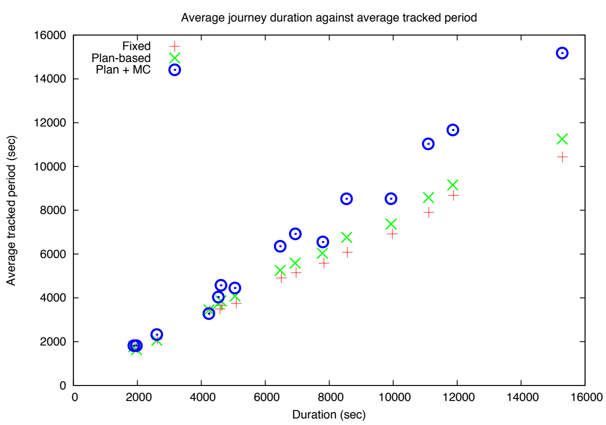

Figure 3: Average time over which the target was tracked to destination against the average journey length.

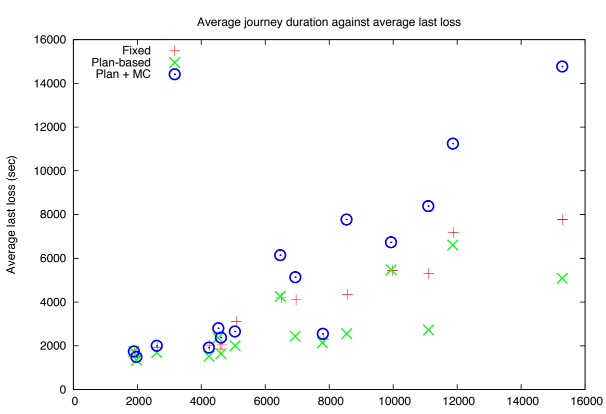

Figure 4: Average time of the last loss of the target against average journey length.

currently use tools based on it to plan SAR efforts. In particular, the SAROPS (Search and Rescue Optimal Planning System) system (Kratzke, Stone, and Frost 2010) has been used by the US Coast Guard since 1974 for SAR operations involving lost objects at sea. SAROPS is composed of two main subsystems: (i) the simulator, and (ii) the planner. The simulator produces a time-dependent PD for the target location using MC particle filtering. Based on the target PD and a collection of available search and rescue units (SRUs), the planner assigns one rectangular search pattern (lawnmower) to each SRU, which then proceeds to execute the pattern. Each SRU executes only one pattern and there is no routing of a vehicle from one pattern to another. The planner seeks to maximise the probability of discovering targets by placing the rectangles intelligently. If no SRU finds the target, the simulator generates a new PD by incorporating information about motion (drift) and about the previous unsuccessful search and then the planner generates a new set of rectangles. This loop is not automated, but it involves the presence of a human who coordinates the two systems and supervises the process.

The SAROPS system has a number of similarities with our technique. As in SAROPS, we also use MCS to develop a prior PD for the target's location, although we do not regenerate the PD at each step. In contrast to our approach, the SAROPS planner works with a one-step lookahead horizon and does not incorporate any long term strategic reasoning. While SAROPS provides a method for carefully orientating the patterns to cover the search area, we concentrate on organising a sequence of patterns to be explored over time. It would be interesting to compare our strategy against SAROPS, but at the time being this comparison cannot be performed since SAROPS is not in the public domain.

Theoretical approaches. Originally, the two areas of searching and tracking were considered separately. Over the last ten years, however, the field of probabilistic SaT has evolved rapidly and a unified approach to SaT has emerged (Furukawa et al. 2006). The probabilistic approach to SAT relies on the use of Recursive Bayesian Estimation (RBE) techniques that recursively update and predict the PD of the target location over time, under the assumption that the prior distribution and the probabilistic motion model of the target are known (Bourgault, Furukawa, and Durrant-Whyte 2003). Although (Bourgault, Furukawa, and Durrant-Whyte 2004) discuss a number of possible constraints that can impact the target motion model (obstacles, force fields and terrain), the target is usually assumed to be subjected to external disturbances and not to move on the basis of its own intentions. Probabilistic-based SaT has proven successful for problems involving stationary targets or targets moving in small geographical areas, simple motion models, static search spaces and short-term missions. However, when these assumptions are not satisfied, RBE techniques perform poorly due to the high computational cost of accurately maintaining a large state space that includes all the possible positions of the moving targets.

Lately, modern approaches to planning with incomplete state information, which are based on Partially Observable Markov Decision Process (POMDP), have been applied to SaT, both for single targets (Hsu, Sun, and Rong 2008) and multiple targets (Bertuccelli and How 2006). In (He, Bachrach, and Roy 2010), the authors present an online, forward search, planning-under-uncertainty algorithm for the road constrained target-tracking problem. In this work, the agent's belief of each target's pose is represented as a multi-modal Gaussian belief and this parametric belief representation is exploited to compute the distribution of posterior beliefs after actions are taken. This analytic computation allows the planner to search deeper by considering policies composed of multi-step action sequences. Deeper searches are beneficial as they result in keeping the targets well-localised for longer periods. This technique has proven successful for small geographical areas, but has not been tested yet on larger regions.

9 Conclusions and Future Work

In this paper, we have presented a hybrid approach to autonomous SaT that combines temporal planning with Monte Carlo simulation. This approach affords a number of benefits. Thanks to the use of sequences of search patterns, which individually can be of any size and orientation, we are able to inspect large and heterogeneous geographical areas. At the same time, since we use automated planning to generate these sequences of patterns, our approach is capable of building mission plans over long temporal horizons. Our method can be easily adapted to a variety of target behaviours and environments. This is because the generation of the initial set of candidate search patterns from which the planner chooses those to execute is based on the target PD, which is obtained by running MCS. Whereas the planning mechanism remains the same, several hypothesis can be incorporated in MCS to reflect different characteristics of the target motion model and the environment, which in turn will produce different initial distributions. Our method can also be easily extended to deal with multiple searchers and multiple targets. In these scenarios, the initial PD will represent probable locations of all the targets and the planning mechanism will take care of assigning different patterns to different searchers in order to maximise the probability of rediscovering the targets. For the sake of planning, search patterns performed by one search unit can be treated as obstacles by the other search units so as to avoid collisions between the searchers. Finally, another advantage of a plan-based approach to SaT is that the behaviour of the observer is predictable and well understood. A plan can be used as a common medium of exchange between the UAVs and the human observers, allowing safer interaction between the drones and other air traffic.

In future work, we intend to construct a different formulation of the planning model, which will allow us to take in consideration the results of unsuccessful searches when choosing the sequence of patterns to execute. In particular, we would like to be able to change the rewards of the patterns based on information concerning which patterns have failed to rediscover the target in the past. In addition, we plan to implement a mixed-initiative framework in which a human operator can modify or recommend a plan or interfere with the execution of a plan. In such a framework, our approach could also be used to provide estimates of success for user generated plans. Finally, we would like to run more extensive experiments to compare our approach with other methods, such as SAROPS.

References

Benton, J.; Coles, A.; and Coles, A. 2012. Temporal Planning with Preferences and Time-Dependent Continuous Costs. In *Proceedings of the Twenty Second International Conference on Automated Planning and Scheduling (ICAPS-12)*.

Bernardini, S.; Fox, M.; Long, D.; and Bookless, J. 2013. Autonomous Search and Tracking via Temporal Planning. In *Proceedings of the 23st International Conference on Automated Planning and Scheduling (ICAPS-13)*.

Bernardini, S.; Fox, M.; and Long, D. 2014. Planning the Behaviour of Low-Cost Quadcopters for Surveillance Missions. In *Proceedings of the 24st International Conference on Automated Planning and Scheduling (ICAPS-14)*.

Bertuccelli, L. F., and How, J. P. 2006. Bayesian forecasting in multi-vehicle search operations. In *AIAA Guidance, Navigation, and Control Conference (GNC)*.

Bourgault, F.; Furukawa, T.; and Durrant-Whyte, H. F. 2003. Optimal Search for a Lost Target in a Bayesian World. In *Field and Service Robotics*, volume 24 of *Springer Tracts in Advanced Robotics*. Springer Berlin. 209–222.

Bourgault, F.; Furukawa, T.; and Durrant-Whyte, H. F. 2004. Process Model, Constraints, and the Coordinated Search Strategy. In *Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA 2004)*, 5256–5261.

CSAR. 2000. *Canadian National Search and Rescue Manual*. Department of National Defence.

Edelkamp, S., and Hoffmann, J. 2004. PDDL2.2: The language for the classical part of the 4th international planning competition. In *Proceedings of the 4th International Planning Competition (IPC-04)*.

Furukawa, T.; Bourgault, F.; Lavis, B.; and Durrant-Whyte, H. F. 2006. Recursive Bayesian Search-and-tracking using Coordinated UAVs for Lost Targets. In *Proceedings of the 2006 IEEE International Conference on Robotics and Automation (ICRA 2006)*, 2521–2526.

He, R.; Bachrach, A.; and Roy, N. 2010. Efficient planning under uncertainty for a target-tracking micro-aerial vehicle. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, 1–8.

Hsu, D.; Sun, W.; and Rong, L. N. 2008. A Point-Based POMDP Planner for Target Tracking.

IMO. 2013. *International Aeronautical and Maritime Search and Rescue Manual (IAMSAR)*. United States Federal Agencies.

Koopman, B. 1946. Search and screening – operations research evaluation group report 56. Technical report, Center for Naval Analyses.

Kratzke, T.; Stone, L.; and Frost, J. 2010. Search and rescue optimal planning system. In *Proc. of the 13th Conference on Information Fusion*, 1–8.

NATSAR. 2011. *Australian National Search and Rescue Manual*. Australian National Search and Rescue Council.

Stone, L. D. 1975. *The Theory of Optimal Search*. perations Research Society of America.