Incorporating Heuristically Generated Search Patterns in Search and Rescue

Helen Wollan

Master of Science

School of Informatics

University of Edinburgh

2004

Abstract

Search and Rescue operations rely heavily on generic search patterns when using aircraft in a search effort. These search patterns need to be manually modified to accomodate the area to be searched. This dissertation proposes to use heuristic methods to generate search patterns to fit the area instead of modifying a generic pattern. With this aim in mind, this dissertation describes an implementation of one such heuristic method in a Search Management tool. This Search Management tool incorporates the ability to view and maintain the status of a search effort along with the search pattern generation. These functionalities provide a tool to use throughout a search effort to define the search area, generate the search patterns as necessary, and view the current status of the search effort. Additionally this Search Management tool includes the capability for XML messaging, allowing users in the field to send updated information to the search manager through the Search Management tool.

Acknowledgements

My thanks to my supervisor Austin Tate, Clauirton Siebra, who helped me figure out I-X and OpenMap, and Peter Wollan who assisted me with the math formulas.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Helen Wollan*)

Table of Contents

Chapter 1

Introduction

Search and Rescue operations are performed throughout the world by both government agencies and agencies in the private sector. These can include looking for a missing child, a hiker who did not return to camp as expected, or even a military aircraft lost at sea. While some may feel that using computers in search and rescue efforts is merely a way to distance the Search and Rescue manager from the operation [Stoffel et al., 1998], looking at the Search and Rescue techniques as a whole, there are a few areas where computers can assist the Search and Rescue coordinators in performing their job more efficiently. One area I will be investigating in this paper is the use and creation of search patterns in search efforts using aircraft.

Search operations are generally completed in the following steps: determining the area to be searched, performing a quick search of the most likely areas, searching the area using search patterns, and finally using area saturation to completely explore the area. Of these steps, the definition of the search patterns is the one requiring the least subjective judgement to arrive at an appropriate solution for the area. In fact, since most search managers use basic, generic patterns that must then be modified to fit the situation, a computer tool may be able to provide a better pattern for a given area.

These search patterns describe in detail the path an aircraft should take when searching an area for any sign of the lost or injured subject. However, these patterns can be limited as the search managers tend to use a select few generic patterns and modify them individually to accommodate the area that needs to be searched. As the problem of finding a subject within an area is very similar to the problem in robotics of finding an object within a polygonal area, heuristic methods from that field of study used to define a pattern can be applied to the Search and Rescue field. These heuristic methods may provide more efficient paths to use in a search, but at the very least they would provide a search path that would not need to be individually modified for each sector that aircraft aircraft search. This would save the search manager time and effort in choosing the generic search pattern to use and modifying it to fit the sector.

Once these heuristic methods have been examined, an implementation of these heuristics would help the Search and Rescue field apply this change to its operations. The search managers would be more willing to use the heuristically generated patterns if a computer application formulated them according to the heuristic methods instead of the manager having to manually follow an algorithm to create them.

By drawing on the information given by Search and Rescue manuals from the UK [Stoffel et al., 1998], Canada [Canada SAR Manual, 1998] and the USA [United States SAR Manual, 1991], this paper delves into the use of patterns in Search and Rescue operations. It then investigates the use of algorithms previously used primarily in other areas such as robotics [Gewali and Ntafos, 1998, Lee et al., 1996, Moret et al., 1997, Ntafos, 1992, Tan, 2001, Tan and Hirata, 2003] and CAD/CAM [Arkin et al., 2001, Arya et al., 2001], and their applications to the generation of search patterns that can be used in searching an area with aircraft. These algorithms detail methods to cover a specified region using tracks of a defined width and based on a variety of related NP-Hard to NP-Complete problems. Using these algorithms to define search patterns that cover an area may provide more efficient solutions than modifying a generic pattern to fit the situation.

Once a selection of these algorithms has been examined, this paper discusses the algorithms and software needed to define these patterns in a spherical space, representing the globe of the earth, and the process of implementing such algorithms in a manner that search managers would be able to use in a Search and Rescue operation. Finally, this paper discuses further areas that such an implementation can be directed to further assist Search and Rescue managers in their efforts.

Chapter 2

Background

2.1 Search and Rescue techniques

Search and Rescue is the act of searching for, rescuing, or recovering by means of ground, marine, or air activity any person who becomes lost, injured, or is killed while outdoors or as a result of a natural or man-made disaster [Stoffel et al., 1998]. This term describes two separate functions which have been developed in different manners. Rescue utilises proven procedures along with a high degree of technical skill for victim retrieval while search for the lost or injured subject has developed into a sophisticated science involving a great many modern investigation techniques. Statistics, probability, human behaviour, interviewing, terrain evaluation and tracking are a few of the standard tools used in a modern search [Stoffel et al., 1998]. While both the search and the rescue aspects of Search and Rescue operations are interesting areas, we are primarily investigating the search half of Search and Rescue.

2.1.1 Definition of the Search Area

The first step in an orderly approach to search strategy is to establish an appropriate search area, or the total area to be investigated. The periphery of the search area is defined using confinement tactics based on the lost subject's behaviour and good initial information gathering techniques such as interviewing. This search area may be redefined if the existing area has been searched to satisfaction with no success or as new information or clues become available. The Probability of Area (POA), or the probability that the subject is in a particular segment of the search area, is defined using the point the subject was last seen and the last known position of the subject. The last known position may be the same as the point last seen, but it could also be a departure point such as a trailhead, campsite, or a discovered clue that can be reasonably verified to be associated with the subject. The last known position changes with time as new clues are discovered while the place that the subject was last seen usually remains constant. The search area does not become smaller but the search effort does grow closer to the subject with every change in the last known position [Stoffel et al., 1998].

There are four major methods to establish a search area. The theoretical search area is a plotted zone on the map indicating the maximum distance the subject could have travelled from the last known position, point last seen, or from a suspected point of departure during the elapsed time. An individual can theoretically travel in any direction from the initial planning point, point last seen, or last known position and the search area is drawn with this in mind.

The statistical search area is derived from previous incidents, as statistics can give a likely distance the subject may have travelled. For example, given data from previous lost hiker incidents in a similar region the tendency may be for the hikers to be found approximately 1.6 miles from the last known position. Information such as this can be used to calculate zones of probability where there is a tendency for the subject to be.

Subjective search areas are usually determined by an experienced search coordinator. Factors affecting subjective search areas include “likely spots” or features which would offer some attraction to the lost person, natural barriers and terrain features, physical clues left by the subject, historical data of the area from case histories, intuition based on experience and special circumstances, and physical and mental limitations of the subject. For the most part these factors are intangible and in the absence of agreement, one authority or personality may strongly influence the perception of the probable area. However, when the last known position is not entirely relevant to the search area, consideration of these factors may prove invaluable.

Search areas may also be defined by the use of deductive reasoning in which the search manager looks at general facts and circumstantial evidence and logically deduces probable conclusions that are not obvious or were not known initially [Stoffel et al., 1998]. This is also a highly subjective method to defining search areas and is often used in conjunction with other methods to define search areas.

Search areas can be established using any or all of the above methods, depending on the preferences of the search manager and which methods the situation warrants. The important point is that the search manager can justify his choice in establishing the search area and that there is a high likelihood that the subject is within the area.

2.1.2 Definition of Search Sectors

The search manager, by using combinations of the methods discussed above, must assure that there exists a defined search area. Once this is clearly defined, the next task is to divide the area into sectors that can be reliably searched. This is done to assure complete coverage of the search area, enable the searchers to complete shift objectives in a reasonable time, and to reduce the overall effort by the searchers.

Defining the sector boundaries requires careful thought, good map reading and, ideally, knowledge of the region. Sectors intended to be searched with ground resources may require different sector boundaries than sectors intended to be searched via the air. In ground search, the use of lines such as latitude and longitude is not ideal as the searchers in the field rarely have the ability to use these lines as identifiable boundaries. Therefore, the choice of boundaries must be based on what can be seen and readily identified in the field by all searchers. Suitable boundaries, for example, may be man-made such as fences or roads, natural such as ridge lines or vegetation breaks, or improvised such as compass lines or point-to-point, also referred to as line of sight [Stoffel et al., 1998]. This is where a person stays within sight of a certain point, a building for example, and conducts the search around this point. This could also be staying within sight of a longer object, such as a road or a fence, when searching and thus expanding the sector while maintaining a reference as to where the searcher is.

In an air search, sector definition can make use of arbitrary lines such as latitude and longitude. There are a variety of methods to define sectors using latitude and longitude. Canada prefers using the GEOREF method for defining sectors [Canada SAR Manual, 1998]. This method is based on a map with a scale of 1:500,000 printed with each GEOREF grid square (1 degree latitude by 1 degree longitude) labelled with a two-letter code. Thirty-minute grid lines are also provided, subdividing each one-degree by one-degree area into four sub-areas. These are identified numerically and referred to as primary squares. These can further be divided into secondary squares labelled alphabetically from A to D. See Figure 2.1 for an application of the GEOREF method.

The USA, however, prefers not to use the GEOREF method. Instead,

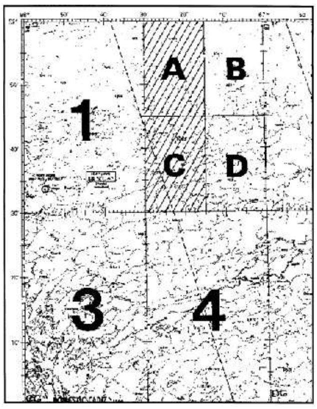


Figure 2.1: GEOREF grid

they suggest other methods to define sectors such as the boundary method, corner method, center point method, track line method and the grid method [United States SAR Manual, 1991]. These methods are merely other ways to use latitude/longitude coordinates to describe an area, often in more concise manners to reduce any radio transmissions required in referring to the search sector. The boundary method describes a rectangular area oriented east-west or north-south by listing the two latitudes and two longitudes. For example, an area may have boundaries 26N to 27N and 64W to 65W. The corner point method can be used to describe any non-circular area by stating the latitude and longitude or geographical features of each corner of the area. The center point method is convenient for describing any regular search area by giving the latitude and longitude of the center point and the search radius, if the area is circular, or the direction of the major axis and the applicable dimensions, if the area is rectangular. The trackline method method lists the start and finish points of a track and the width of the coverage area around that track. This is often used if the search manager knows the path the subject took and is concentrating the search around this path. The grid method is based on grid charts and is very similar to the GEOREF method as it, too, refers to grids using details listed on a grid chart [United States SAR Manual, 1991]. This method requires everyone working with the same grid chart, as any differences can cause confusion when referring to the sector.

Another important aspect to consider when defining sector boundaries is the size of a sector. A sector should be coverable by a search team in a four to six hour period. This allows the team to complete the assignment, have a break, and be shifted to another sector while also allowing them a sense of accomplishment in finishing an objective. Sectors that are too small can cause logistic problems with moving the search teams too often [Stoffel et al., 1998].

2.1.3 Sector Probabilities

After defining the search sectors, the search manager needs to assign probabilities to the sectors. These probabilities represent the probability the subject is in that sector and therefore ranks the segments in the order of priority that each should be searched and how thoroughly the the sector should be searched. A sector with a low probability should be ranked low in the priority list of sectors and not necessarily searched as thoroughly as a sector with a high probability. These values can be changed throughout the search as new sectors are added or clues are found. There are many techniques to assign probabilities to sectors such as the Mattson Consensus Approach and the Sector Ladder Technique [Stoffel et al., 1998].

The Mattson Consensus Approach entails the principal members of the search planning team to individually and independently assign values to each sector, totalling 100 points over all sectors. These values are then averaged by sector, with these averages representing the probabilities of the sectors [Stoffel et al., 1998].

The Sector Ladder Technique is a simple method to prioritise the sectors. The sectors are ranked in order of priority with the rest of the world, or any area outside the defined search area, last. As sectors are searched, they drop down in the ranking and the unsearched sectors move up [Stoffel et al., 1998].

The probability assigned to a sector may change as clues are found. If a highly relevant clue is found, the relevance being a subjective decision made by the search manager, then the probabilities assigned to each sector should be re-evaluated and the sectors re-prioritised.

The probability that the subject is in a given sector is just one issue facing the search manager. The probability of detecting the subject within the sector must also be considered. This probability is based on several factors including the method of searching the sector, the size of the sector, and the size, experience, and motivation of the search team [Stoffel et al., 1998]. The search manager's goal should be to maximise the probability of detection with the available resources. In doing this, the search manager needs to consider several factors that can impact the probability of detection. One such factor is searcher fatigue, which can have a vital impact on the probability of detection. While data on the fatigue and effectiveness of ground searchers is lacking, maritime and air search and rescue fatigue studies have shown that searchers are most effective during the first four hours with the maximum efficiency within the first hour [Stoffel et al., 1998]. Other factors include the time available or allowed to accomplish the search, the size of the sector to be searched, the search method and the number of passes or sweeps to be performed when assigning resources to search a sector. The search manager needs to balance the probability the subject is in the area with the resources available as well. For example, the likelihood of success is greatly increased with the higher number of sweeps made on a sector [Stoffel et al., 1998, Canada SAR Manual, 1998]. However, additional sweeps require additional resources so a sector with a low priority in the ranking would not have many sweeps dedicated to it.

2.1.4 Search Methods and Patterns

Once the search manager has defined the search area, divided it into prioritised sectors and assigned the resources required for each sector, he needs to define the search method for each sector. There are three basic search methods. The first is a method used when the search team arrives shortly after the subject is reported missing. Small teams are dispatched to check adjacent trails, ridges, drainages, ponds and all known structures in the area. Roads are also checked and patrolled if possible. The second search method uses search patterns, which are used to cover the sectors rapidly and efficiently with the search teams being slightly larger. The third search method uses area saturation to completely explore a sector [Stoffel et al., 1998]. The first and third methods are fairly straightforward, so we are concentrating on the second method: the definition of search patterns.

Search patterns are vitally important to the second search method as they describe how thorough the sector will be covered. Ground searchers tend to use less structured search patterns than aircraft searchers. One example of a ground search pattern is a search team consisting of three searchers and one compass-bearer. The compass bearer follows a designated bearing and the searchers are free to wander, checking likely spots while continually guiding on the compass bearer [Stoffel et al., 1998]. This can be seen in Figure 2.2 where the compass bearers are represented by the 'X' lines and the searchers follow random paths such as the 'O' lines, investigating turf at their own will. Ground searchers following techniques such as this can easily be paired with aircraft search following one of the structured patterns described below. This combination of

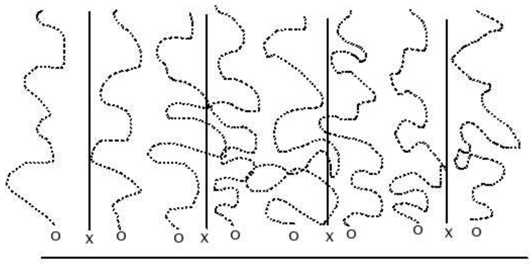


Figure 2.2: Typical Ground Search Method

ground and air search can fully and rapidly search a given sector of the search area.

Search patterns for aircraft tend to be more structured and also provide varying degrees of coverage. There are several different generic search patterns which are commonly used in aircraft search operations. These include creeping lines, expanding squares, sector search, and contour search [Stoffel et al., 1998, Canada SAR Manual, 1998, United States SAR Manual, 1991]. Pilots are generally assigned a sector to search with the pattern, altitude, and speed determined by the search manager. The search manager chooses the pattern based on several constraints. If the search area is large, the creeping lines pattern (Figure 2.3) is most suitable since there are fewer turns and navigation is more accurate.

The expanding square search pattern (Figure 2.4) is used when the location of the search subject is known with reasonable accuracy. It requires precise navigation to avoid gaps in the coverage and can thusly cause searcher fatigue more quickly than with other patterns.

The sector pattern (Figure 2.5) is used when the search area is not extensive and the subject is difficult to detect. The chief advantage of the sector search is that the

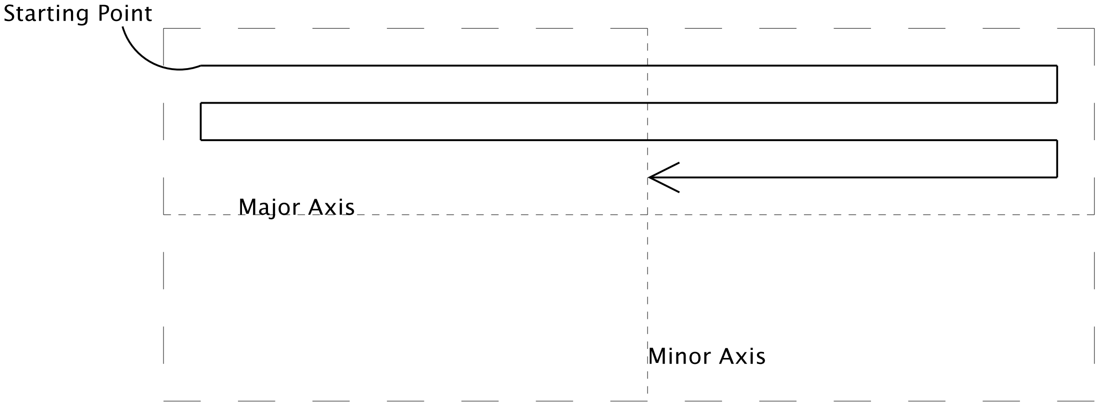


Figure 2.3: Creeping Line search pattern

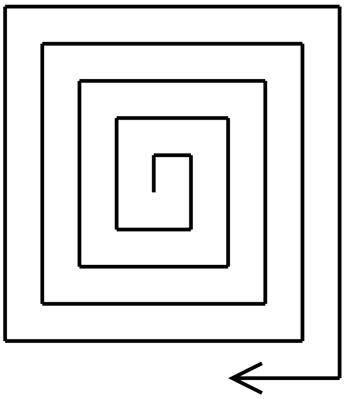


Figure 2.4: Expanding Square search pattern

track spacing at the center of the search is very small, resulting in a greater probability of detection in the area of greatest probability, assuming the center of the sector has the highest probability.

A contour search (Figure 2.6) is one that follows the contours of the land,. This can be a hazardous search procedure so the aircraft used must be highly manoeuvrable and the crew must be experienced in mountainous terrain. Also, only one aircraft may be assigned to a sector for a contour search to minimise the risk

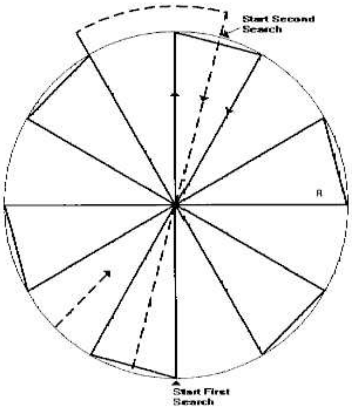


Figure 2.5: Sector search pattern [Canada SAR Manual, 1998]

[Canada SAR Manual, 1998]. However, this search technique is the best choice in mountainous terrain as it allows the aircraft to maintain a consistent distance from the earth, increasing the likelihood of finding the subject.

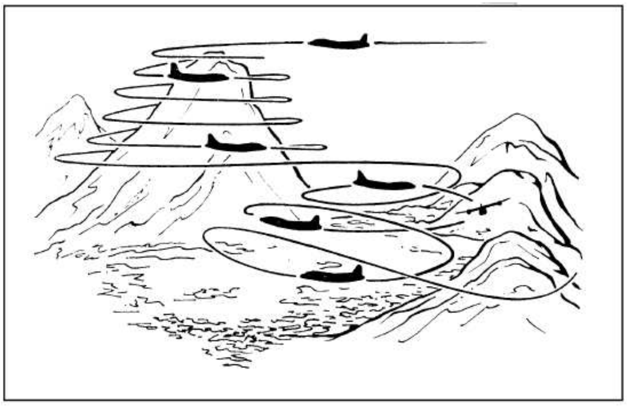


Figure 2.6: Contour search pattern [Canada SAR Manual, 1998]

When defining the search pattern to use in searching a sector, some additional factors need to be decided. These include the sweep width and track spacing. The sweep width is a measure of the detection capability of the aircraft. The speed, altitude and type of aircraft, weather, visibility of the subject and several other variables affect the sweep width. The sweep width is obtained by choosing a value less than the maximum detection range of the aircraft so that scattered targets may be detected beyond the sweep width are equal in number to those which may be missed within the sweep width. The track spacing is the distance between adjacent search tracks, whether these are by simultaneous sweeps of several units or successive sweeps of a single aircraft. The smaller the track spacing is, the higher the likelihood will be of detecting any object within the sector. However, decreasing the track spacing increases the time for any given aircraft to cover the search area, or alternatively requires more aircraft units to complete the search in the same time. Figure 2.7 shows how the sweep width and track spacing relate to each other can be calculated while Figure 2.8 shows the relation between track spacing and track width.

2.2 Search Pattern Heuristics

The generic search patterns used by aircraft, as described above in Section 2.1.4, may not cover the sector in the most efficient manner. A method used in robotics to search a given polygonal region for an unknown subject can be applied to the Search and Rescue area. These methods are primarily based on the Lawnmower problem [Moret et al., 1997]. The Lawnmower problem, an NP-Hard problem, requires the inside of a polygonal region to be completely covered at a minimal cost, typically with the shortest path. Related problems include the Ice Rink problem [Moret et al., 1997], the pocket-milling problem [Ntafos, 1992], and the d-Sweeper problem [Ntafos, 1992]. In the milling problem, one also requires that a cutter never cut outside the boundary of the region, and thus never approach the boundary to within the cutting radius. This characteristic is less important in search

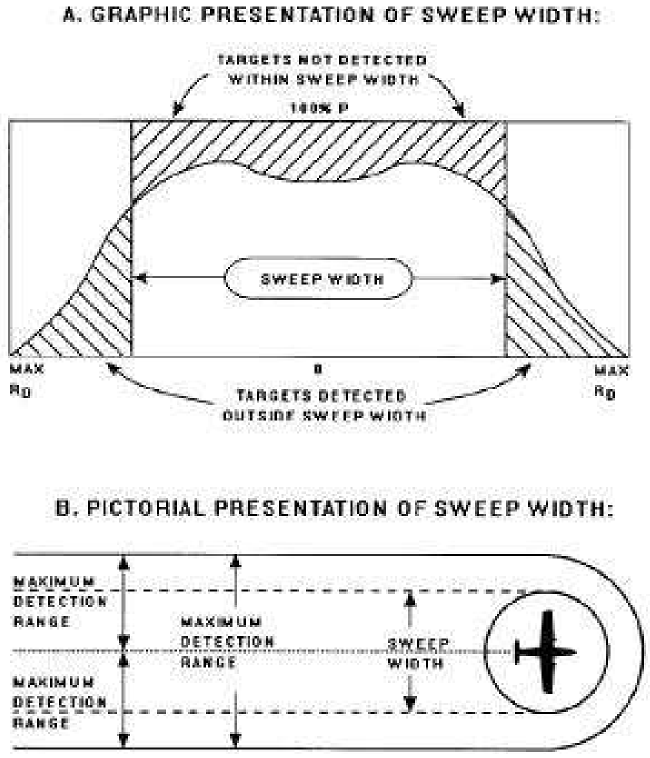

A. GRAPHIC PRESENTATION OF SWEEP WIDTH:
TARGETS NOT DETECTED WITHIN SWEEP WIDTH
100% P
SWEEP WIDTH
MAX R_D
0
MAX R_D
TARGETS DETECTED OUTSIDE SWEEP WIDTH
B. PICTORIAL PRESENTATION OF SWEEP WIDTH:
MAXIMUM DETECTION RANGE
MAXIMUM DETECTION RANGE
MAXIMUM DETECTION RANGE
SWEEP WIDTH

Figure 2.7: Sweep Width Calculation [Canada SAR Manual, 1998]

patterns as it is usually acceptable to go outside the sector boundaries, however minimising these forays outside the sector is important for efficiency. While this problem is NP-Hard, there are several heuristic algorithms to find approximated solutions [Moret et al., 1997, Ntafos, 1992, Gewali and Ntafos, 1998, Arya et al., 2001, Tan and Hirata, 2003, Arkin et al., 2001, Lee et al., 1996, Tan, 2001]. These algorithms provide near-optimal routes that provide efficient coverage of a polygonal area. As there are several different algorithms to provide solutions to these related problems, I will be covering a few most relevant to the pattern development in Search and Rescue.

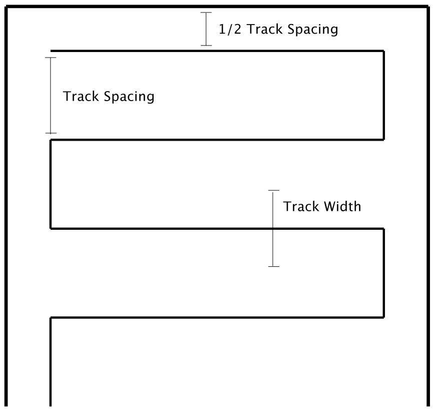


Figure 2.8: Track Spacing and Sweep Width

2.2.1 Ice-Rink Problem

Moret [Moret et al., 1997] suggests two algorithms to find efficient coverage of polygonal areas that can be used by airplanes in a search operation. These algorithms minimise the number of turns as airplanes take more time turning than continuing in a straight line. The algorithms also attempt to keep the radius of the turns as large as possible to maximise the speed of the aircraft in a turn as well. Both algorithms are based on a Zamboni polygon [Moret et al., 1997] which is defined as a simple polygon composed of a rectangular middle piece flanked by two end caps, each consisting of a polygonal line monotonic with respect to the two end segments of the rectangle. Any convex polygon is a Zamboni polygon since we can regard it as composed of two end caps with a degenerate middle piece.

The first such heuristic algorithm, called the Zamboni Decomposition algorithm, directly decomposes the polygonal area to be searched into a variation of Zamboni polygons. Since we are more interested in generating large pieces then generating few pieces, we can use greedy algorithms to decompose the polygonal area with a local iterative improvement, attempting to increase the size of the largest Zamboni piece identified before proceeding recursively with the remaining pieces. This ensures we get the largest Zamboni polygons we can find in the polygonal area. Each Zamboni shape then yields a number of sweep tracks, generally in a creeping line pattern. The algorithm then uses a Travelling Salesperson Problem (TSP) solution to connect the tracks optimally, knowing the cost of each connection [Moret et al., 1997]. See Figure 2.9 for an example of the sweep tracks the Zamboni Decomposition algorithm forms for an irregular polygon.

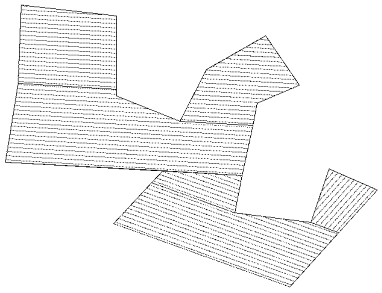


Figure 2.9: Zamboni Decomposition [Moret et al., 1997]

The second heuristic algorithm based on the Zamboni polygon, the Uniskeletal Decomposition algorithm, uses the characteristic of Zamboni polygons as a definition of their middle piece by a medial axis. This is the polygonal line used in the translation. After computing the skeleton of the polygon, with some noise-reduction methods to avoid detailed branches induced by small features of the perimeter, the algorithm decomposes the skeleton into generalised polygonal lines. Each polygonal line defines a generalised version of a Zamboni polygon, which can be swept by paralleling the polygonal line. This contour-following approach is commonly used in the milling problem. Then, as with the Zamboni Decomposition, the algorithm uses a TSP solution to connect the sweep lines [Moret et al., 1997]. See Figure 2.10 for an example of the sweep tracks the Uniskeletal Decomposition algorithm forms for an irregular polygon.

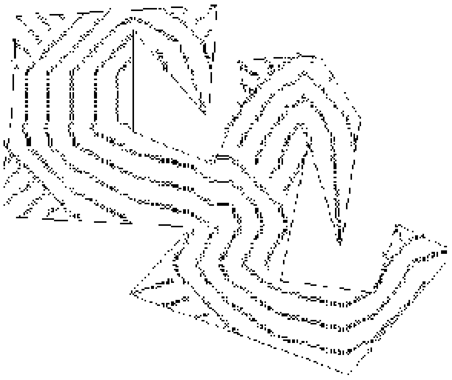


Figure 2.10: Uniskeletal Decomposition [Moret et al., 1997]

Both of these approaches concentrate on the core of the search area and lend themselves well to an optimisation based on maximising early returns. Both approaches also allow the incorporation of additional information or constraints, such as a required sweep edge or line. This enables the route created by the algorithms to account for searching terrain specifically, such as searching along the edge of a forest or along a coast. However, both approaches also suffer from the creation of small artefacts, or very small regions of the polygon that require a large number of short tracks and a large number of turns for very little area, this being more pronounced in the Uniskeletal Decomposition approach. This can be countered by re-aligning some of the sweep lines to conform to the sweep pattern used in the adjacent, and usually larger, area. The TSP algorithm can also be prioritised under the reasonable assumption that the target is unlikely to be found in the last few percent of the area.

2.2.2 *D*-Sweeper Problem

Ntafos [Ntafos, 1992] refers to the d-sweeper problem to solve a similar problem. The d-sweeper problem is based on the Watchman Route problem, which asks for the shortest route from a point back to itself with the property that each point in a given space is visible from at least one point along the route. However, this problem assumes the visibility inside the polygon is infinite, which causes this problem to become a question of finding the route around the boundary of the polygon within a certain distance. Since, in Search and Rescue, visibility becomes a problem, we turn to the d-sweeper problem, which asks for the shortest route from a point back to itself when there is a visibility range d and we are interested in viewing the whole interior of the polygon. The d-sweeper problem is also related to the TSP problem in grids. It is possible to superimpose a grid of unit size $2d$ on the polygon, clip portions of the grid on the exterior of the polygon and ask for the shortest TSP route that visits all vertices of the grid. Then we can adapt the grid solution to the polygon by making local adjustments [Ntafos, 1992]. The quality of the resulting solution depends on how fine the grid is. If the resulting route visits each vertex on the grid only once, we have a Hamiltonian circuit and an optimum TSP route and therefore an optimum search pattern. An example of the d-sweeper method on a pyramid-type shape can be seen in Figure 2.11. This shows the outline of the polygon with a TSP solution inside the polygon. This solution is built on a grid, covering all vertices of the grid within the polygon.

These algorithms are examples of heuristic solutions to the NP-Hard problem of finding efficient routes that cover a given polygonal area. These solutions could be applied to a Search and Rescue operation to devise search patterns that efficiently cover a sector of the search area. The generic search patterns that are currently used in Search and Rescue may not use the resources of the pilot and aircraft very efficiently because

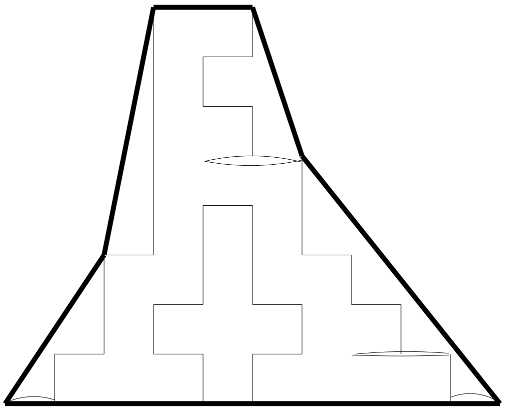


Figure 2.11: *d*-sweeper pattern within a polygon

the overall shape of the search pattern could easily not match the shape of the sector to be searched. By using the resources more efficiently, the search of the sector could be completed with fewer resources, which allows additional sweeps of a sector given the available resources and thusly increases the probability of detection.

It would be harder to apply these more efficient search patterns to ground search as there are some learnability issues. These routes may not be the most obvious routes to use in searching an area and would therefore not be easy for the searchers in question to follow carefully and search the sector thoroughly. Also, it is more difficult for a person on the ground to maintain a direct heading as there may be terrain obstacles in the way or just the common human nature not to walk in a straight line for an extended period of time. Therefore, it is better to apply these new search patterns to aircraft search.

Chapter 3

Implementation of a Search Management Tool

3.1 Objectives

One primary aim of this project is to implement an automatic generation of search patterns. This automatic generation of search patterns can work together with other project goals to provide a Search Management tool that provides functionality for a search manager to coordinate various aspects of the search operation. To do this, the resulting implementation needs to fulfil some basic objectives.

3.1.1 User Defined Search Area

One of the first steps a search manager does when beginning a search operation is to define the area to be searched. This area is then divided into sectors based on various probabilities that the subject is in an area and to define the terrain of the area. For example, there could be a high probability that the subject is in a certain field with lower probability the subject is in an adjoining forest. These probabilities are determined based on a number of aspects, detailed in Chapter 2. To assist the Search and Rescue manager in defining these sectors and therefore defining the search area, which is wholly and completely composed of sectors, the implementation will allow the user to "draw" the search sectors onto an underlying map using the mouse. Once drawn, the search manager can get the latitude/longitude coordinates of the vertices of the sectors to use in further tasks in managing the search operation.

The Search Management tool will make a few assumptions on the sectors the search manager will draw on the map. The sectors will be in a closed polygonal shape with no "holes," which are smaller polygons completely contained within a larger one. The sectors will also neither overlap nor leave gaps in the search area. In other words, any point in the search area will be covered by one and only one sector. If an area of the map is not within a drawn sector, it will be assumed to be outside the defined search area for this search operation.

3.1.2 Defining Constraints on Sectors

Once the sectors have been defined by the search manager based on the assumptions, the next step for the search manager is to define various constraints for each sector. These constraints include the probability the subject is within the sector and the individual searchers assigned to the sector. This Search Management tool will include the ability for a search manager to assign probabilities and individual searchers to the sectors. The tool will also provide a general "catch-all" area for the search manager to store information for each sector. This stage of the definition allows the search manager to add any additional notes, reflections, or important information regarding each sector.

3.1.3 Automatic Generation of Search Patterns

After the search manager has defined the sectors and any constraints he feels necessary for the sector, the Search Management tool will automatically generate the search patterns. These patterns will be aimed towards air search as the automatic definition of air search patterns has the most potential usefulness in a Search Management tool. These patterns will be defined in latitude/longitude coordinates and will give efficient coverage of the search sector based on the user-defined track width, or the visibility distance of the aircraft. This visibility distance depends on the altitude the aircraft is flying at, the current weather conditions, and any other factors that may impact the ability of the aircraft pilots to search the sector. If the user has not defined a track width then a default value will be assigned. These search patterns will be based on the heuristic algorithms described in Section 2.2 as these patterns are more likely to be more efficient then the generic search patterns that search mangers tend to use today.

3.1.4 Status Reporting

A search manager not only needs to define the search area, sectors, and search patterns, but he also needs to monitor the status of the various sectors. The Search Management tool will provide the capabilities to store the percentage a sector has been searched, the probabilities of the subject being in the sector, and any other notes the searchers may have on the sector. These notes may include vital information such as an item of the subject was found in a given sector or any “hunches” the individual searchers may have in a sector. This status keeping functionality would not be quite so beneficial if someone had to monitor a phone continuously and transcribe any new status information. Instead, the Search Management tool will provide capabilities for the individual searchers to send information to the main tool using basic XML messaging commands.

Also, users at the main tool can send information regarding the sectors to the individual searchers using XML messaging. This information includes the vertices of the sectors, the vertices of the search patterns, and any status information available on the sectors.

3.2 Tools for Implementation of Search Management

Instead of creating this implementation from scratch, I investigated using existing technologies to meet the objectives above. Since searches need to be planned on a geographical map, a Geographical Information System seems to be the most logical technology to use in a search management implementation.

3.2.1 Geographical Information Systems

Geographical Information Systems (GIS) combine layers of information about a place to give the user a better understanding of that place. A GIS provides improved management of resources by allowing the user to query, analyse, and map data to support a decision making process [GIS website, 2004]. To compare this to a common system used, one could look at transparencies on an overhead projector. A map of the area would be the first transparency laid on the projector. Additional information could be placed on separate transparencies. For example, the political outlines of countries could be on one transparency and if placed on top of the map, the audience could see both the underlying map and the political boundaries of the countries. In this manner, one can add additional layers of information to the map and remove layers that are unnecessary. A GIS simply takes this idea and uses it in a computer application. In a GIS, layers can be either opaque or transparent. Opaque layers do not allow the users to view underlying layers and can be useful when the application needs the underlying layers but the user does not wish to view those layers. Transparent layers allow the user to view all the information below the layer along with the transparent layer. While this can make the map cluttered if several layers are active, it also allows the user complete control of the information in the map.

3.2.1.1 Why use a GIS in Search Management?

A search manager needs to monitor several aspects of the Search and Rescue operation as well as to be able to define constraints on the map. All this information could be very messy if it were all displayed directly onto a map. Having several text areas, menu options, or dialogs boxes would also not be ideal in this situation. Therefore, storing the information on various layers in a GIS would provide a concise and easily viewable method to display the information a search manager needs to be able to access in a search operation. If the search manager does not need to view the search patterns, he can simply deactivate the pattern layer and view the status instead. The relevant data is still in the application, just not visible to the user. Also, having this information displayed on a map provides the search manager an easy way to match the information in the application with the visual aspects of the area. For example, a list of coordinates for a search sector, while providing all the necessary information for the search manager, does not provide a very good way to visualise the sectors so the search manager can ensure the entire search area is covered.

3.2.1.2 Which GIS to use?

There are several GIS applications in the market, the most popular being ESRI's ArcGIS [ESRI ArcGIS, 2004]. While ArcGIS would provide all the functionality required in a search management implementation, the cost can be prohibitive.

Several other companies provide GIS applications as well including MapInfo Inc. [MapInfo, 2004], and BBN Technology [BBN OpenMap, 2004]. While these GIS applications would provide the functionality required for this implementation, I chose to use BBN Technology's OpenMap because it is both open source and written in Java. Being able to write the layers required for the Search Management tool in a language I am familiar and comfortable with reduces the learning curve in this project while open source allows me the ability to investigate in high detail how the GIS works if I encounter any problems.

3.2.2 Communication Tools

After choosing a GIS application to base the Search Management tool on, I turned to finding a communication tool that would be able to handle the XML messages described in the objectives of this implementation. Ideally the communication tool would have the ability to integrate with the GIS to easily manage the messages sent and received. I-X [Tate et al., 2004] seems to be a logical choice in this manner as it has already been integrated with BBN's OpenMap GIS by Clauirton Siebra, a PhD student at the University of Edinburgh [Siebra, 2004], and includes the ability to send messages in basic XML encapsulating relevant information that can be stored on the implementation of OpenMap.

3.2.2.1 I-X

I-X, by using process panels, is used to support individual users who are carrying out processes and responding to events in a cooperative working environment [Tate et al., 2004]. These panels support the tracking of personal or group issues, the planning and execution of activities and the checking of constraints. These panel can also be connected to a Map Tool, displaying information in the layers of OpenMap as necessary. I-X, while already containing the integration with BBN OpenMap, also provides messaging capabilities using XML messaging commands. These enable the users to send information via messaging protocols from one I-X panel to another, having the relevant information appear on the Map Tool when the message arrives.

3.3 Steps to Implementation

Once the objectives have been set and the supporting applications have been chosen, the next step is to divide the tool implementation into steps that can be individually implemented. An obvious division of this search management implementation is to first enable the user to draw the sector definitions on the map. Once this is completed to satisfaction, the next step is to provide the functionality for the user to define the status information. These include the probability the subject is within each sector, the searchers assigned to each sector, the percentage each sector has been searched, and any further notes the search manager wishes to store attached to the sectors. The final step is to implement automatic generation of the air search patterns in each sector. This step automatically generates a list of latitude/longitude vertices that, when connected, provide the search pattern an aircraft should use when searching a particular sector. These automatically generated patterns are based on the Zamboni Decomposition algorithm described in Section 2.2.1.

3.4 Algorithms Needed for Implementation

Creating the Search Management tool requires more than just simple coding. Described here are several algorithms and equations that require additional explanation in how they work.

3.4.1 Point in Polygon

This algorithm provides the means to decide if a given point is inside a closed polygon. This is used several times in the Search Management tool to decide if a latitude/longitude coordinate is inside a user-defined sector, or polygon. While there are a few algorithms to decide this, the simplest is to draw a line through the point in question along the latitude line and count the number of times the line crosses an edge of the polygon on either side. If the number of times that line intersects an edge on both sides of the point is odd, then the point in question is inside the polygon. Otherwise, it must be lying outside the polygon [Finley, 1998]. There are additional cases handling the instances when the point is on an edge or vertex of the polygon. Before counting the number of times the line crosses an edge of the polygon, first ensure that the point is not a vertex. If the point is on an edge of the polygon then the number of times the drawn line will pass through the polygon on one side will be zero, and on the other side it will be an odd number. Using this property, it is easy to determine if a point lies on the edge of the polygon. An example of this algorithm can be seen in Figure 3.1. In this figure, the point in question has the line drawn through it, intersecting the edges of the polygon. On the right side, the line intersects the polygon three times while on the left side it intersects the polygon five times. Since both of these are odd, the point must be contained within the polygon.

3.4.2 Azimuth Calculation

Another very important algorithm in the Search Management tool is the azimuth calculation. The azimuth, also referred to as the bearing or direct heading, is the an-

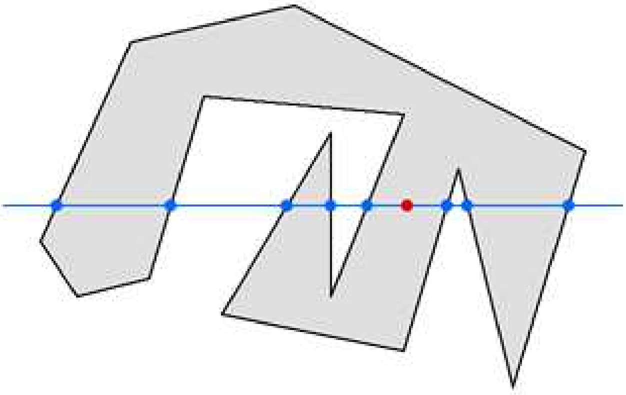


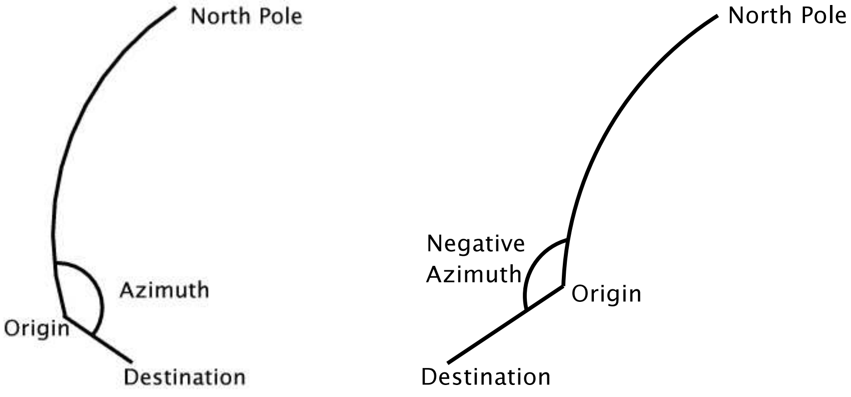
Figure 3.1: Example of Point In Polygon

gle between the north pole and an arc between two latitude/longitude coordinates. The azimuth can also be thought of as the slope of a line segment between two latitude/longitude coordinates as it is a way to determine which direction a line is moving on a spherical shape. As the Search Management tool uses this function to calculate the angle between to edges of the polygon, the initial inclination is to use a simple vector dot-product or other equations from geometry and trigonometry. However, I soon realised this cannot be done as latitude/longitude coordinates lie on the earth, which is not flat. Therefore, geometric equations based on a planar space would not suffice in this situation. Instead, we had to turn to spherical geometry and the formula [Veness, 2002]:

$$Azimuth = \tan\left\{\frac{-\sin(long1 - long2) * \cos(lat2)}{\cos(lat1) * \sin(lat2) - \sin(lat1) * \cos(lat2) * \cos(long1 - long2)}\right\}$$

where $lat1$ and $long1$ are the latitude and longitude of the originating point, specified in radians. $Lat2$ and $long2$ are the latitude and longitude of the destination point, specified in radians.

This formula gives the angle measurement between the arc from the origin to the north pole and the origin to the destination. See Figure 3.2(a) to better explain this.



(a) Positive Azimuth angle (b) Negative Azimuth angle

Figure 3.2: Azimuth angle calculation

However, if the arc segment from the origin to the destination is heading from east to west, the azimuth calculation is negative instead of being greater than π as shown in 3.2(b). Using this formula, and implementing cases, it is possible to calculate the angle between two arcs on the earth.

3.4.3 New Point Given an Azimuth and Origin

To create the search pattern inside the sectors we have to manipulate the azimuth calculations to get the azimuth bisecting the interior angle of the sector and heading inwards on the polygon. To calculate a new coordinate along this azimuth at a certain distance from the origin would be a simple calculation in a planar world using slopes and basic geometry, but can be difficult, or at least more complex, in a spherical world. This equations to calculate this new point on a sphere are as follows [MAPINFO-L, 2004]:

$$\begin{aligned} DestLat &= \arcsin\{\sin(lat1)*\cos(dist)+\cos(lat1)*\sin(dist)*\cos(azimuth)\} \\ dLong &= \tan\left\{\frac{\sin(azimuth)*\sin(dist)*cos(lat1)}{\cos(dist)-\sin(lat1)*\sin(lat1)}\right\} \end{aligned}$$

$$\begin{aligned} interLong &= (long1 - dLong - \pi) * (long1 - dLong + \frac{\pi}{2}) \\ &\qquad \text{if } interLong < 0 \text{ then } interLong = interLong + 2\pi \\ DestLong &= interLong - \pi \end{aligned}$$

The $lat1$ and $long1$ represent the latitude and longitude of the origin and are both represented in radians. The $azimuth$ must also be in radian units for this formula to work. The $dist$ represents the distance from the origin we wish to plot the new point. This, too, is represented in radians, which is easily calculated by converting the distance into nautical miles and multiplying by $\frac{2\pi}{360*60}$ as one nautical mile is equal to one minute of an arc, or $\frac{1}{60}$ of a degree. The new coordinate, $dist$ from the origin, is at the point $(DestLat, DestLong)$.

3.4.4 Integrating Algorithms and Equations to generate Search Patterns

By using the equations and algorithms described above we were able to generate search patterns in a given sector using an algorithm based on the Zamboni Decomposition algorithm discussed in Section 2.2.1. The first step in this algorithm is to examine the sector, or polygon, to check if it is a convex polygon. If it is convex, or having all internal angles less than π radians, then the shape is already a zamboni polygon and we can generate the search pattern directly. If the sector is not convex then it needs to be divided into a series of non-overlapping convex polygons. For example, if the original sector contains some obtuse angles, or internal angles greater than π, the polygon will be divided. To do this, the algorithm compares the distance from the offending vertex, the one with internal angle greater than π, to the other vertices of the polygon, avoiding the adjacent vertices. It then draws a line from the offending

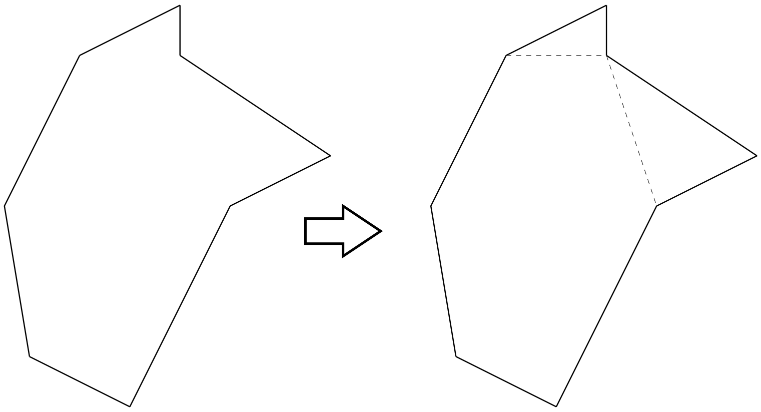


Figure 3.3: Splitting an irregular polygon into concave polygons

vertex to the closest vertex, forming two polygons. The algorithm then checks both of these polygons for obtuse angles and if any are found then the process repeats until no internal angles are greater than π. This process can be seen in a simple example in Figure 3.3.

Once the sector is divided into purely convex polygons, the next step is to generate the search pattern to use within the sector. This is done individually for each convex polygon and then the search patterns for the individual polygons are connected using a simple greedy algorithm.

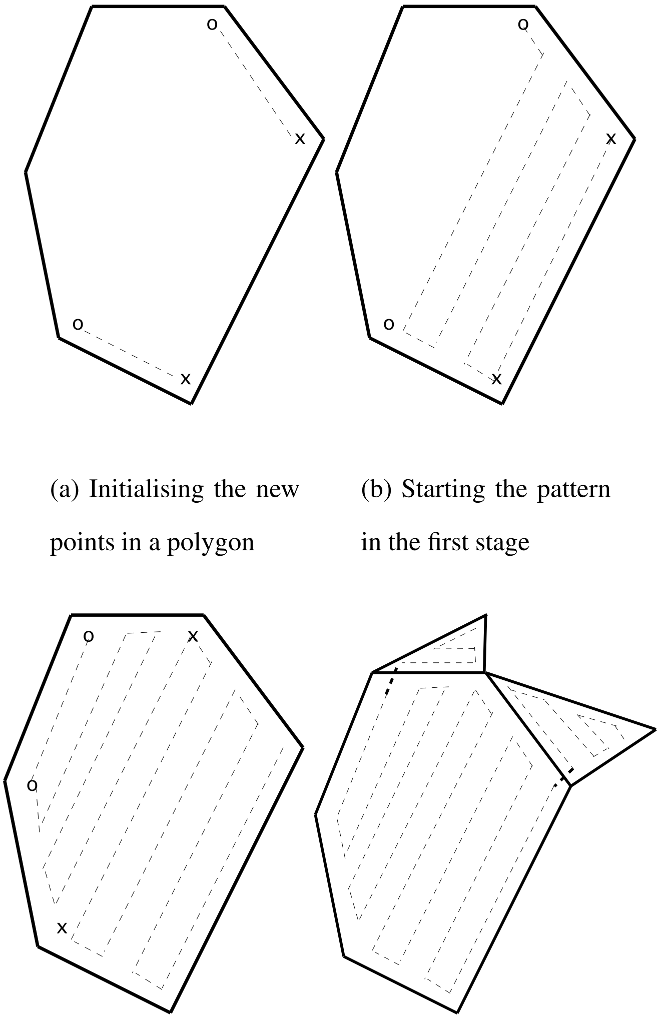
To generate a search pattern for a polygon, the algorithm first calculates the longest edge of the polygon to use as the baseline of the search pattern. From the two vertices on this line, the algorithm calculates the bisecting azimuth of each vertex and the new points along these bisecting azimuth inside the polygon. These new points are the user-defined track width away from the vertices. Because the angle is less than π, the new point will be less than the half the track spacing from the sides of the polygon, ensuring correct coverage of the entire polygon. New points such as these are also computed for the adjacent vertices as well, as in Figure 3.4(a) where the x's mark the interior vertices of the longest edge and the o's mark the adjacent internal vertices. Once these points are calculated, the next step is to calculate the azimuths along the adjacent edges as the dotted lines indicate in Figure 3.4(a).

Using these azimuths it's possible to continue to calculate new points along them at a distance of the track spacing. This is continued, connecting the track lines at alternating edges as in the creeping line search pattern (Figure 2.3) until one of the new points calculated is at or beyond the calculated interior point of one of the adjacent vertices, as in Figure 3.4(b). At this point, the next pair of vertices around the polygon are bisected and the interior points are calculated as in Figure 3.4(c) and the pattern building is continued as before. Once the polygon is completely covered with the track lines any additional polygons in the sector have their patterns defined in a similar manner as in Figure 3.4(d).

The final step in the search pattern generation is to connect the individual patterns of the polygons inside a sector to make one continuous search pattern. To do this, we simply calculate the shortest distance between a start and an end point of the various individual patterns and greedily connect them. The thick dotted lines in Figure 3.4(d) show these connecting lines and the final result is a continuous search pattern that ensures coverage of the entire polygon with a user-defined track spacing.

3.5 Implementation of Search Management tool

The actual implementation of the Search Management tool took several steps to completion. The first such step was to enable the user to draw sectors onto the underlying map in BBN OpenMap. The next step was to add the pattern generation including the functionality to split the irregular polygons into convex polygons. The final such step



(a) Initialising the new points in a polygon

(b) Starting the pattern in the first stage

(c) The search pattern for a convex polygon

(d) The search pattern for an irregular polygon

Figure 3.4: The steps to creating search pattern tracks

was to enable the storing of the sector information. The results of these steps, along with the problems encountered during implementation, are discussed below.

3.5.1 Drawing the Sectors

The first step in the Search Management implementation was to add the functionality for drawing sectors on the map. This entailed grabbing the mouse click on the map and converting the x/y coordinates into latitude/longitude coordinates for storage. However, since the user may wish to click on the map without necessarily drawing the sectors, I needed a way to turn the drawing on and off. To reach this goal, I added a properties button to the layer. By pressing the button 'Start Building', the user can indicate he is ready to draw the sectors on the map by clicking on the vertices of the sectors and double clicking to indicate the final vertex of the polygon. At a double click, the tool checks if the drawn polygon is closed and if not, it automatically closes the polygon. The tool then adds this sector to I-X as a constraint, allowing it to be referred to by other layers. When the user is finished drawing the sectors and wishes to turn off the drawing capabilities, he merely navigates to the properties button for the Sector Definition layer again and presses the 'Finish Building' button. In Figure 3.5, a sector is in the process of being drawn on the map. The button to finish the building is displayed as well.

I encountered an unexpected problem when adding this functionality to the tool. The integration of BBN's OpenMap and I-X includes a 'World Objects' layer that allows the user to add objects to the map by right-clicking on the map. Because this layer also interprets the mouse events on the map and is of a higher priority than the Sector Definition layer I defined, the mouse events were being filtered through the World Objects layer. This had the surprising result that the double-clicks were not

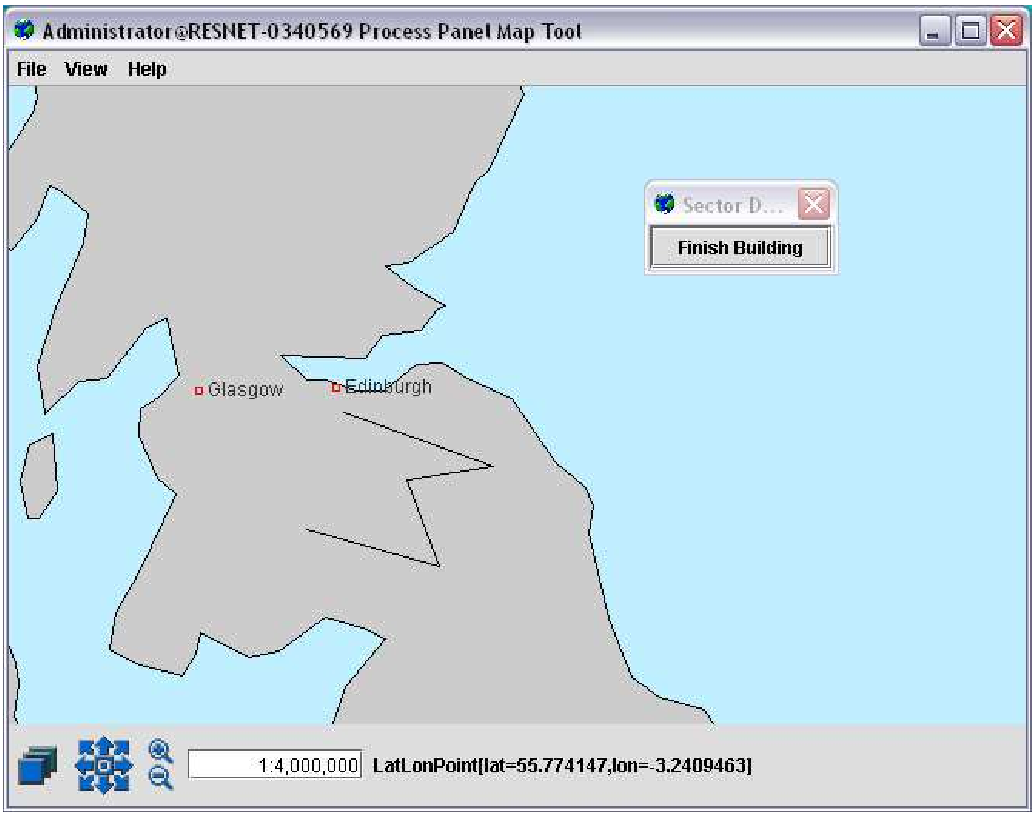

Administrator@RESNET-0340569 Process Panel Map Tool
File
View
Help
Sector D...
Finish Building
Glasgow
Edinburgh
1:4,000,000
LatLonPoint[lat=55.774147,lon=-3.2409463]

Figure 3.5: Drawing of Sector in progress

being communicated to the Sector Definition layer. As a workaround, I disabled the World Objects layer upon starting the tool. However, the user can still enable the layer and cause the problem to occur. A permanent solution to this problem would be to either modify the World Objects layer to allow all mouse events through or to completely remove the World Objects layer from the user.

3.5.2 Splitting Polygon and Pattern Generation

Once I had the ability to draw sectors on the map and the method described in Section 3.4.4, I was able to work on the layer to define the patterns in the sectors. However, I quickly ran into a problem when I realised even given the azimuth calculations (Section 3.4.2), it is difficult to calculate the internal angle of a vertex in a polygon. It is fairly simple to calculate an angle between two intersecting arc segments, but determining if that angle is the internal or external angle of the polygon is a different matter. After giving this some thought, I decided to handle this using cases.

The first step is to determine the sign of the azimuth calculations of the arcs composing the vertex of the polygon. Depending on their signs, both positive, both negative, or one of each, we can calculate the bisecting azimuth of the obtuse side of the vertex accordingly. For both positive or both negative azimuths, the bisecting azimuth is $\frac{azimuth1-azimuth2}{2}$ where $azimuth1 > azimuth2$. For azimuths not of the same sign, the bisecting angle calculation is more complex. If the positive azimuth, *azimuth1*, is greater than $\frac{\pi}{2}$, the bisecting azimuth is $2\pi - \frac{azimuth1-azimuth2}{2}$. However, if the positive azimuth, *azimuth1*, is less than $\frac{\pi}{2}$, the bisecting azimuth is $\frac{azimuth1+azimuth2}{2}$.

Using these cases, we now have the bisecting azimuth of the obtuse side of the vertex. To determine if this obtuse side is to the interior of the polygon, we first calculate a point along the bisecting azimuth using the formula in Section 3.4.3. With this new point, we check if it is in the polygon using the algorithm described in 3.4.1. If the point is inside the polygon, the obtuse angle is to the interior of the polygon. However, if the point is not inside the polygon, the obtuse angle is to the exterior of the polygon.

If the vertex has an obtuse angle to the interior of the polyton, as decided above, we then split the polygon using the method described in Section 3.4.4 above. With the two polygons now created, we can continue checking the interior angles of the polygons, splitting as necessary, until there are no more obtuse angles to the interior of a polygon.

Using the similar cases to the ones above, we can calculate the bisecting azimuth of the vertices in the sub-polygons to generate the patterns using the method described in Section 3.4.4. Figure 3.6 shows a pattern generated for a convex sector. This pattern completely covers the polygon given the sweep width defined, in this case the sweep width is 10 km. The search patterns are displayed in the Search Pattern layer, allowing the search manager to easily display the patterns or not.

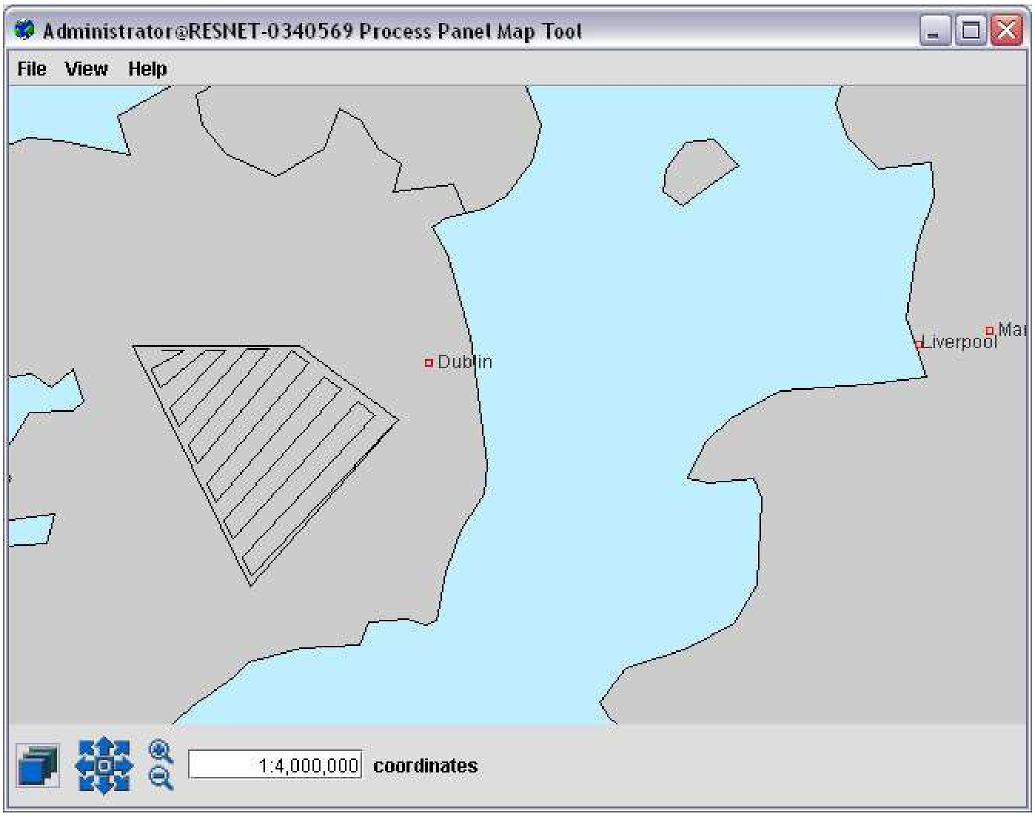


Figure 3.6: Generated pattern for convex sector

The search pattern becomes more complicated when the sector defined is not a simple convex polygon. An example of the search pattern generated for a more complex polygon can be seen in Figure 3.7. This search pattern shows three individual patterns that are then joined to create one more elaborate pattern, again using a sweep width of 10 km.

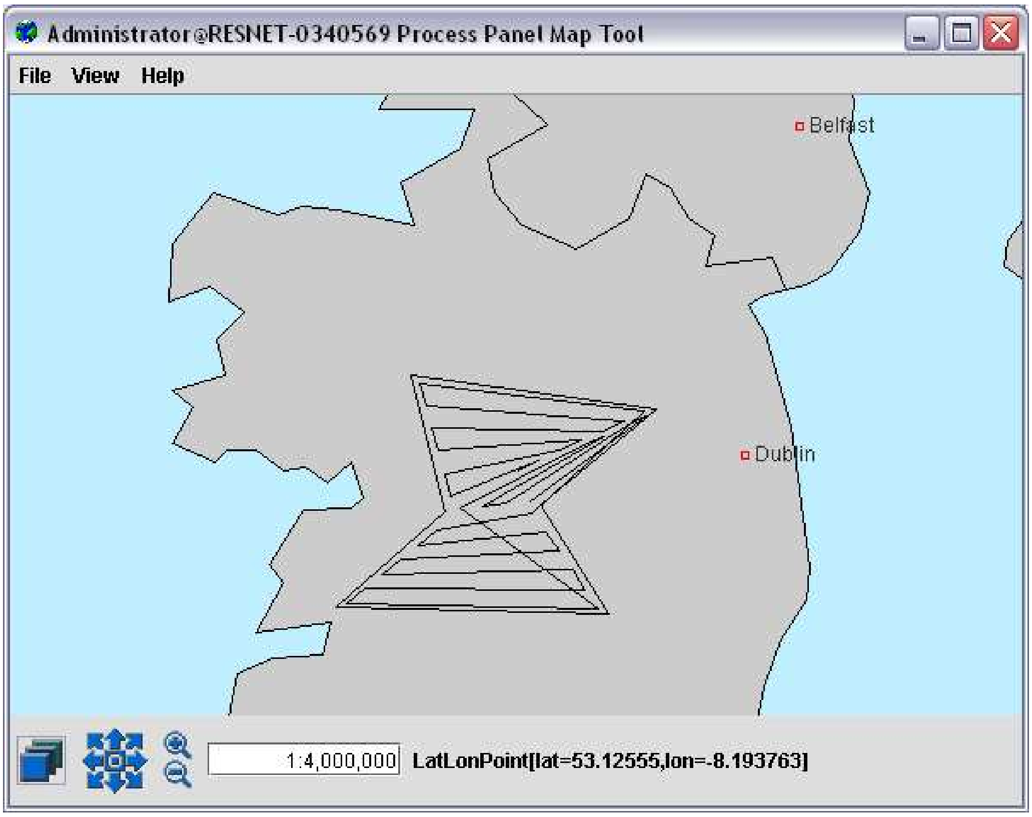

Administrator@RESNET-0340569 Process Panel Map Tool
File
View
Help
Belfast
Dublin
1:4,000,000
LatLonPoint[lat=53.12555,lon=-8.193763]

Figure 3.7: Generated pattern for a complex sector

3.5.2.1 Outstanding Issues with Pattern Generation

Unfortunately, due to the complex nature of the pattern generation aspect of the Search Management tool, there are some outstanding issues. Due to the spherical nature of the earth, relying on the azimuth to determine the direction can cause some problems. Namely, over large distances the azimuth value can change while maintaining the same direction. The Search Management tool does not account for this so therefore, if the sector is large the generated pattern may not be complete. To overcome this problem the Search Management tool needs to continually re-calculate the azimuth between the interior points, as in Section 3.4.4. Since most search managers would not be using sectors large enough to cause this problem, I decided not to correct it for this project.

The Search Management tool also does not optimally split the complex polygons into smaller polygons. The algorithm merely splits the complex polygons into smaller convex polygons which can then have a pattern generated in. However, some of the smaller polygons can sometimes be merged and still have a search pattern generated successfully for this larger polygon. This would require an additional step in the method discussed in Section 3.4.4 to find and merge these polygons.

Finally, the Search Management tool does not optimally connect the individual search patterns in a complex polygon. Currently it compares two individual search patterns at a time and optimally connects them, but does not consider all of the search patterns at once. To correct this would necessitate an additional step after all the individual search pattern have been defined to optimally connect them using a simple Travelling Salesman algorithm.

3.5.3 Status of Sectors

Maintaining and displaying the status of the sectors is the final step in the implementation of the Search Management tool. This is done using two layers in the tool. The menu options, shown in Figure 3.8, give the user the ability to view and update the probability the subject is in the sector, the status of the search in the sector, the individual searchers assigned to the area, and any additional notes. These options, aside from viewing the individual searchers, are available in dialog windows. Figure 3.9 shows the dialog window for the displaying and creating additional notes the search manager may have on the particular sector. These notes are timestamped, allowing the search manager to easily see the timeline on which the notes were created. The other menu options bring up similar dialog windows to allow the search manager to view and modify the values.

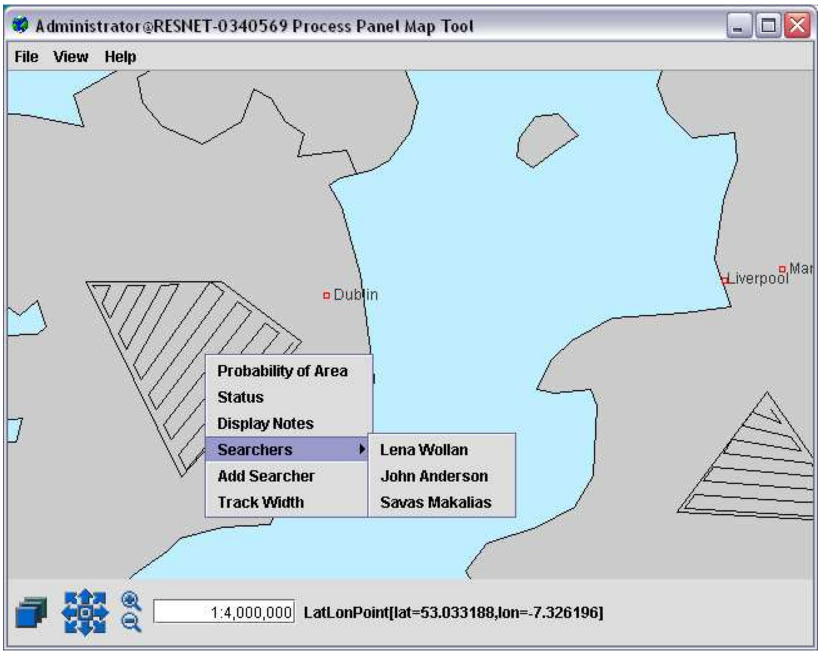
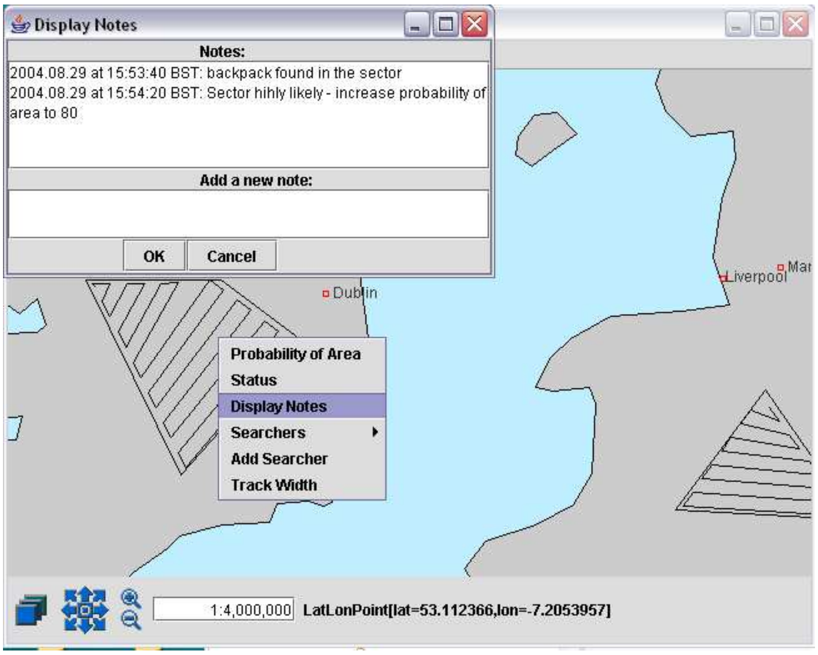


Figure 3.8: The menu options for each sector

These menu options are primarily visible through the Sector Definition layer in the Search Management tool. However, the tool also includes a Status layer. This layer provides the colour-coding of the sectors with regards to their status, or percentage searched. If the percentage searched is between 0 and 33, the sector is coloured red; if the percentage searched is between 34 and 66, the sector is coloured orange; if the percentage searched is between 67 and 99 the sector is coloured yellow; and finally the sector is coloured green if it is 100 percent completely searched. This simple colour coding allows the search manager to see at a glance which sectors have yet to be searched, which ones are in progress and at what level of completion they are at, and which ones have been completely searched. Figure 3.10 shows several sectors in varying stages of search completion. The Search Pattern layer has been disabled to



Display Notes

Notes:

2004.08.29 at 15:53:40 BST: backpack found in the sector
2004.08.29 at 15:54:20 BST: Sector hihly likely - increase probability of area to 80

Add a new note:

Figure 3.9: The Display Note dialog window

give an uncluttered view of the sector status, however the patterns would be visible as all layers in the Search Management tool are transparent. This figure shows one sector that is still in the beginning stages of the search, one sector that is partially searched, one sector that is nearly completely searched, and two sectors that have been searched to completion. From this view, the search manager would be able to determine the search effort still requires a great deal of work as a majority of the search area is not completely searched or nearing completion.

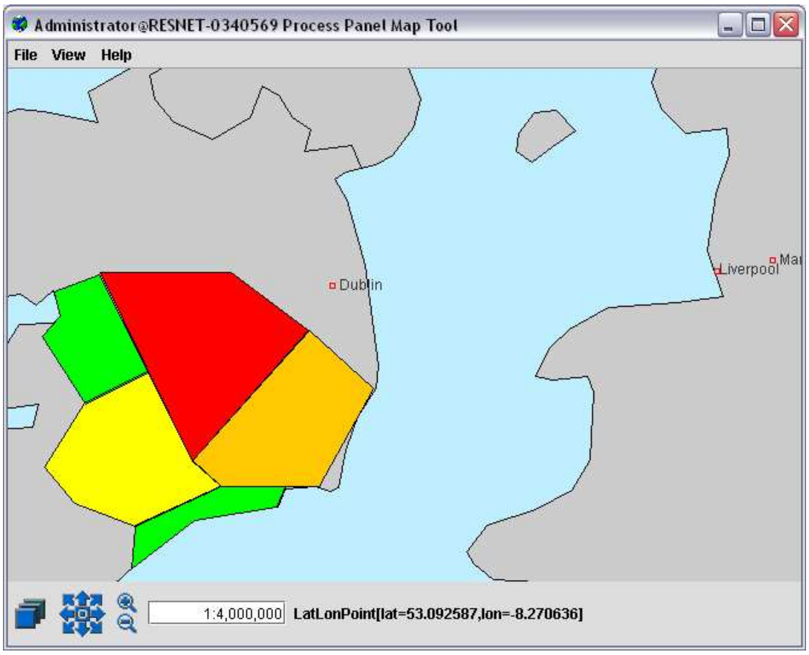

Administrator@RESNET-0340569 Process Panel Map Tool
File
View
Help
Dublin
Liverpool
Mai
1:4,000,000
LatLonPoint[lat=53.092587,lon=-8.270636]

Figure 3.10: Several sectors with varying search statuses

Chapter 4

Conclusions

The project detailed in this paper had several aims to satisfy. These included investigating the use of search patterns in Search and Rescue operations and the implementation of automatic generation of search patterns for search managers to use. With the first aim in mind, we examined the use of generic search patterns based on the definition of sectors and the probabilities assigned to those sectors. These search patterns, however, need to be modified on a case-by-case basis to ensure coverage of the individual search sectors. Because this adds a large amount of overhead in terms of time and energy on the part of the search manager, we investigated other possibilities. The Robotics and CAD/CAM fields often also require covering an area with complete coverage. The Robotics field often gives robots the task of searching an area for a particular object. This involves searching the area within the visibility of the robot until the object is found. CAD/CAM uses this general idea in milling. In milling, a mill needs to cover a defined area completely but not cut outside the area. In both of these fields, the algorithms to define the route used by either the robot or mill try to minimise the distance of the route. Search patterns, and the use of aircraft to follow them, should also be formed with this idea in mind as the shorter the distance overall to cover a sector, the more sweeps of the sector can be performed within a given amount of time and fuel.

There are several different algorithms that find heuristic solutions to this general problem. In Section 2.2, this paper discusses three such algorithms. The first, Zamboni Decomposition, splits the irregular polygon that represents the sector into zamboni shapes, finds the shortest route within these individual shapes, and then connects these individual search routes to compose a larger route that covers the entire polygon. Uniskeletal Decomposition finds the skeletal polygonal lines of the irregular polygon that represents the sector. Using this, it generates search tracks parallel to the line until the polygon is covered. The final heuristic algorithm covered in this paper, the d-sweeper algorithm, places a grid in the polygon and draws basic travelling salesperson routes on the grid and thusly finds a route to cover the polygon. Applying these heuristic algorithms to the Search and Rescue field can help improve the search routes used by aircraft and thereby improve the search effort.

This leads into the next aim of the project, which is to implement a tool which automatically generates search patterns for a search manager. This paper discusses the applications, algorithms, and mathematical equations necessary to generate a search route using the Zamboni Decomposition algorithm for an irregular polygonal shaped sector. Creating a search route usable in Search and Rescue is more difficult than in Robotics or CAD/CAM as the earth's surface comes into play. In both Robotics and CAD/CAM, the algorithm is finding a route on a planar space. However, in Search and Rescue the primary problem arose with the understanding that the earth is an irregular sphere where plain geometry has no bearing. Even the use of spherical geometry can cause some problems because the earth's surface is not perfectly spherical. While the Search Management tool tries to overcome this problem, some issues still exist with the pattern generation (Section 3.5.2.1).

Even with the outstanding issues present in the Search Mangagement tool, the tool generates usable search patterns for most user-defined sectors. These patterns are generated using spherical geometry in a method described in detail in Section 3.4.4. This method involves splitting the user-defined sector into manageable polygons, calculating the azimuths of various points within the polygon, and generating the search pattern using these azimuths.

The tool also stores the status of the sectors along with other information pertaining to the sector. This information gives the search manager a simple way to store pertinent information such as the probability the subject is within the sector, the individual searcher assigned to each sector, any relevant notes regarding the sector, and the percentage of the sector completely searched. The tool not only provides a means to view the percentage complete, but also color-codes the sectors depending on their status. This color coding varies from red to orange to yellow depending on the percentage, and finally green indicates the sector is 100 percent searched. This color coding gives the search manager a quick and easy way to look at the map and determine the overall status of the search effort along with approximate status of the individual sectors.

The combination of the search pattern generation and the status information available make the Search Management tool a useful tool for search managers. It provides a new pattern generation using route generation heuristic algorithms previously used in other areas while also providing an easy way for the search managers to maintain and view the status of the search operation.

Chapter 5

Future Work

There are many tasks the Search Management Tool could be expanded to encompass. These tasks would provide a better tool for the search manager to use but were not feasible objectives in the time allowed.

5.1 Pattern Definition in Pilot Routing Terms

The search patterns currently generated by the Search Management tool are represented in a sequence of latitude/longitude coordinates that are the vertices of the pattern generated. However, pilots rarely use sequences of coordinates to define their flight routes. An appropriate additional functionality of the Search Management tool would be to include a conversion from the latitude/longitude coordinate list to a directly usable format for the pilots. This includes a list of the bearings, or azimuths, and distances to travel along those bearings instead of the coordinate list. With this directly usable list, the search manager could give the pilots the search plan directly instead of wasting time translating the pattern from the list of latitude/longitude vertices to the bearings and distances.

5.2 Terrain Obstacles

The Search Management tool does not account for any terrain features in defining the search patterns or assisting the search manager. Instead, it relies on the search manager to define the search sectors accordingly to avoid troublesome terrain features. Such features can include cliffs, waterfront, and roads. All of these affect the search effort as they affect the probabilities the subject is within an area. For example, if the lost subject is a child it is not very reasonable to have an aircraft search over open water as it's unlikely a child, if in the water, would be visible from the air. With an automatic analysis of the terrain and avoidance of unlikely areas, the search pattern generated would be more efficient in covering the likely areas which is more beneficial to the search effort.

Initially the Search Management tool could allow the user to enter such terrain obstacles on the map as features. This would allow the user to define the important terrain obstacles while ignoring those irrelevant to the situation. However, this adds maintenance work to using the Search Management tool. Because increasing the workload on the search manager is not a goal of the Search Management tool, the tool could take advantage of one of the features of a GIS. GIS applications give the users the opportunity to read information directly from the underlying map, removing the need for the user to predefine terrain obstacles. To do this, however, the GIS requires a map of sufficient detail to provide relevant information at the level a search manager would require. This would remove some maintenance work on the side of the search manager while providing better search patterns for a given sector.

5.3 Search Pattern Selection

Currently the Search Management tool only generates one type of search pattern - a modification of the creeping line pattern using the Zamboni Decomposition algorithm. This, however, may not be the most relevant pattern to use in the given situation; the sector (Figure 2.5) or contour search (Figure 2.6) may be a better choice. As the user, the search manager, is knowledgeable of the sectors to be searched and the generic search patterns commonly used, he would be able to make an educated choice of search pattern if given options. With this in mind, it would be beneficial to give the user a selection of recommended patterns to use on the area with perhaps a default if the user does not choose one. This not only allows the user greater control over the Search Management tool but also would provide an additional functionality that would benefit the search operation in general.

5.4 User-Modifiable Sectors and Patterns

Another feature which would allow the user greater control over the Search Management tool would be to implement the ability for the user to modify the sectors or patterns after creation. Currently, once a sector has been drawn on the map or defined in I-X, the sector cannot be modified. If there is a mistake or if the sectors need to be redefined a sector must be deleted and re-defined. Giving the user the ability to tweak the sectors after drawing them would most likely reduce frustrations with the tool and increase it's usability. Enabling the user to modify the patterns after generation would give the user greater control over the patterns stored in the tool. This would, perhaps, cause the users to follow the suggested patterns while performing their search instead of making modifications while in the search, giving the search manager more control over the actual searches being performed and also an accurate record of the searches completed.

5.5 Automatic Analysis and Suggested Deployment of Resources

Since the search manager can enter the probabilities of the sectors and the people available to perform the searches, it would be a logical extension for the Search Management tool to analyse this information and suggest which resources should be deployed to the various sectors. With the suggested resource deployment, the tool would also automatically generate the search patterns according to this deployment, saving the search manager the necessity of defining this information himself. Of course, the search manager should be able to override any suggestions as other information may come into play that cannot be analysed by the Search Management tool. However, this would save time in the planning stage of the search operation, allowing additional time for the actual completion of the search operation. This is an important benefit when time may be limited or even vital to the subject. Often search management decisions need to be made very quickly and taking some of this decision-making away from the search manager allows time for other tasks to be completed.

5.6 Complete Search and Rescue Management

The Search Management tool discussed in Chapter 3 only covers a small part of the Search and Rescue efforts. Enlarging the tool to handle a Search and Rescue operation from start to finish, would be a logical goal as then the Search and Rescue teams could work completely within one software tool. This package could include the ability to record interviews with the subject's family and friends, analyse the statistical information about the area, and other areas of the search effort. The package could also include the rescue side of Search and Rescue operations. For example, it could store information regarding the availability of resources in an area and suggest any deployment of appropriate resources if needed. This would relieve the coordinators of the Search and Rescue operations the necessity of knowing all the resources in the area including their available skills and current availability to handle emergencies.

The features listed in this chapter are merely suggestions of directions the Search Management tool can be taken in future work. Any of these extensions would increase the functionality and overall usability of the Search Management tool in real world cases.

Bibliography

[Arkin et al., 2001] Arkin, E. M., Bender, M. A., Demaine, E. D., Fekete, S. P., Mitchell, J. S. B., and Sethia, S. (2001). Optimal covering tours with turn costs. ACM-SIAM Symposium on Discrete Algorithms (SODA 2001).

[Arya et al., 2001] Arya, S., Cheng, S.-W., and Mount, D. M. (2001). Approximation algorithm for multiple-tool milling. *International Journal of Computational Geometry and Applications*, 11(3):339–372.

[BBN OpenMap, 2004] BBN OpenMap (2004). http://openmap.bbn.com.

[Canada SAR Manual, 1998] Canada SAR Manual (1998). National search and rescue manual, bga209001/fp001-dfo 5449. http://www.foothills-sar.ab.ca/manuals.html.

[ESRI ArcGIS, 2004] ESRI ArcGIS (2004). http://www.esri.com/software/arcgis/index.html.

[Finley, 1998] Finley, D. R. (1998). Point-in-polygon algorithm - determining whether a point is inside a complex polygon. http://www.alienryderflex.com/polygon.

[Gewali and Ntafos, 1998] Gewali, L. P. and Ntafos, S. (1998). Watchman routes in the presence of a pair of convex polygons. *Journal of Information Sciences*, 105:123–149.

[GIS website, 2004] GIS website (2004). http://www.gis.com.

[Lee et al., 1996] Lee, D., Yang, C., and Wong, C. (1996). Rectilinear paths among rectilinear obstacles. http://citeseer.ist.psu.edu/lee96rectilinear.html.

[MapInfo, 2004] MapInfo (2004). http://www.mapinfo.com.

[MAPINFO-L, 2004] MAPINFO-L (2004). Mi-l new lat/long from distance/bearing. http://testdrive.mapinfo.com/tdc/mapinfo-l.nsf/0/9f5ad6aebf379caf85256ec30066d43f?OpenDocument.

[Moret et al., 1997] Moret, B. M., Collins, M., Saia, J., and Yu, L. (1997). The ice rink problem. Workshop on Algorithm Engineering.

[Ntafos, 1992] Ntafos, S. (1992). Watchman routes under limited visibility. *Computational Geometry: Theory and Applications*, 1:149–170.

[Siebra, 2004] Siebra, C. (2004). personal communication.

[Stoffel et al., 1998] Stoffel, R. S., Jones, D. A. T., Cooper, D. C., Swombow, C., and Andrew, T. (1998). *Search Is An Emergency: A Text for Managing Search Operations*. ERI Publications and Training, one - revised edition.

[Tan, 2001] Tan, X. (2001). Fast computation of shortest watchman routes in simple polygons. *Information Processing Letters*, 77:27–33.

[Tan and Hirata, 2003] Tan, X. and Hirata, T. (2003). Finding shortest safari routes in simple polygons. *Information Processing Letters*, 87:179–186.

[Tate et al., 2004] Tate, A., Dalton, J., de Siebra, C., Aitken, S., Bradshaw, J. M., and Uszok, A. (2004). Intelligent agents for coalition search and rescue task support. Nineteenth National Conference of the American Association of Artificial Intelligence, (AAAI-2004).

[United States SAR Manual, 1991] United States SAR Manual (1991). National search and rescue manual volume 1: National search and rescue system. http://www.hc3.navy.mil/sarmm/pubs.html.

[Veness, 2002] Veness, C. (2002). Calculate distance and bearing between latitude/longitude points. http://www.movable-type.co.uk/scripts/LatLong.html.