ARMY RESEARCH LABORATORY

Terrain Exploration by Autonomous Robots

by Aivars Celmiņš

ARL-TR-2299

September 2000

Army Research Laboratory
Aberdeen Proving Ground, MD 21005-5067

ARL-TR-2299 **September 2000**

Terrain Exploration by Autonomous Robots

Aivars Celmiņš
Computational and Information Sciences Directorate, ARL

Abstract

Terrain exploration with the help of autonomous unmanned ground vehicles can be very cost effective in several military applications, such as mine clearing, decontamination, surveillance, and similar operations. A subtask in autonomous operations is planning the traveling route so that the main goal (the exploration of the terrain) is achieved in the shortest time, the entire area of interest is covered, and the terrain features that are discovered during the exploration are taken into account in real time. In this report, we propose a solution to the subtask. The solution employs a navigation function that is computed using an algorithm based on Huygens' Principle in optics. The proposed solution is an extension of previously reported route-finding methods to predetermined destinations in open terrains. In such terrains, it is necessary to distinguish between terrain in the vicinity of the robot and at farther distances. In the vicinity, the terrain must be carefully examined; therefore, a detailed vicinity terrain map is needed. Terrain features at larger distances can be handled by using approximate representations. In this report, two methods for handling open terrain problems are presented: a vicinity map method and a telescopic map method. The advantages of each method are discussed and examples illustrate how to use the methods in exploration tasks.

Table of Contents

INTENTIONALLY LEFT BLANK.

List of Figures

Figure | Page

1. Introduction

In this report, we consider terrain exploration by autonomous ground vehicles. The advantage of using unmanned ground vehicles for dangerous tasks is obvious; the Army has used unmanned ground vehicles for a number of different military tasks, such as mine clearing, munitions disposal, and range maintenance (Contreras 1998; Moore 1998). The unmanned vehicles are typically operated with remote control by an operator and observer who is located at a distance from the vehicle. The usefulness of the teleoperated vehicles can be considerably enhanced by removing the need for a nearby operator and employing autonomous robotic vehicles. To operate autonomously, a robot must be able, among other things, to plan its route in an unknown or partially known terrain according to given restrictions. For example, one might seek a route for an exploring robot so that by the end of the journey, the whole area of interest has been covered and the travel time has been used efficiently. This route planning task for an exploring robot is similar to the task of finding a route to a given destination through partially known terrain. Finding routes to given destinations has been investigated previously by the author (Celmiņš 1997, 1998, 1999); in this report, the extensions of some of the previous methods to terrain exploration tasks are described.

Autonomous robot navigation presents two different problems, each of which requires its own solution. The two problems are long-range route planning and motion control of the robot for short-range movements (Celmiņš 1997). In this report, we are concerned with the former problem in which one seeks a rough outline of the route that the robot should follow. The negotiation of the planned route is the subject of the second task in which proper motions of the robot are determined, taking into account the conditions of the robot's immediate neighborhood. The details of the path actually traveled generally differs from the planned route because local conditions are not taken into account by the solution of the first task.

The method described here for long-range route planning is a navigation-function method (Rimon and Koditschek 1988, 1989, and 1990). In navigation-function methods, the navigation route is determined in two steps. A first step establishes a navigation function over a two-dimensional terrain plane. The navigation function is defined as having absolute minima only at locations which the robot is tasked to visit, having no relative minima, and depending only on such elements of the terrain information which are relevant for the planned route. The relevance of the information depends on the task assigned to the robot. For instance, the threat level is important only if the robot is supposed to avoid danger, visibility might be important for scouting missions, and navigation speed is important for finding a fast route. The second step of the method consists of finding the navigation route by a local algorithm on the navigation function, such as determining a path of steepest descent.

In this report, we are concerned with the exploration of a specified area in the shortest possible time. Therefore, the salient property of the terrain is the navigation speed. We assume that the property of the terrain is present in the robot's memory in a raster map (the terrain plane is subdivided in quadratic cells and an average navigation speed is specified for each cell). The size of the terrain cells governs the granulation of the input information and the accuracy of the computed route; the proper granulation depends on the navigation task. Thus, to decontaminate a dwelling, a cell size of $10 \times 10\ cm^2$ might be required; for a mine-clearing task in a battlefield, a cell size of $1\ m^2$ might be adequate; for area reconnaissance, $10 \times 10\ m^2$ or larger cells can be sufficient.

The navigation function is computed from the speed map, but because the speed map is granulated, the function is computed only at the centers of the terrain cells. This granulation of the navigation function has numerical advantages because the function can be computed by a very simple and fast algorithm that uses Huygens' Principle in optics. The second step—the computation of the route—is done in the granulated navigation function and the route is presented as an ordered list of terrain cells that the robot should visit.

In battlefield applications, usually the terrain is open; available terrain maps cover a much larger area than is necessary for the navigation task. Because the route planning must be done on-board and in real time, it is important to reduce or eliminate unnecessary map information. In this report, we consider reduction of information by employing a vicinity map that contains detailed relevant information and covers only a limited neighborhood of the robot. As the robot moves, new vicinity maps are generated in sequence so that the route-planning algorithm always uses complete terrain information for a limited number of steps ahead. Information about distant parts of the terrain, which are located outside the vicinity map, is used only to provide a general direction for the planned route.

The next section describes in detail the determination of the navigation function and the computation of optimal routes. Section 3 describes the application of the route-finding methods from section 2 to terrain exploration tasks where the goal is complete exploration of a prescribed area using sensors with finite sensing distances. Section 4 presents application examples of the methods, while section 5 provides a summary of conclusions.

2. Route-Planning With Huygens' Relief

2.1 General Description. Planning of long-range navigation routes usually involves an optimization problem, such as finding the shortest route that avoids obstacles, finding a route that is farthest removed from dangerous areas, or finding the fastest route in general terrain. One of the simplest and most applicable of these problems is finding a route from an initial location R to a destination D with the shortest travel time. (A solution process for such a problem can also determine, after slight modification, routes with the least exposure to danger or with the shortest crossings of wet areas, or similar restrictions.) The optimal route can be found with the help of a navigation function that models signal propagation in a space with variable signal propagation speed. The description of the algorithm starts by considering the problem of finding the path in a given map. An extension of the algorithm to exploration tasks in open and unknown terrains will be presented in section 3.

Let D be the location of the destination and let $v(x,y,z)$ be the signal propagation speed. Then, the travel time of a signal from D to the location R of the robot along a path $P(u) = [x(u), y(u), z(u)]$ is

$$T(P) = \int_{D}^{R} \frac{ds}{v} = \int_{u_D}^{u_R} \frac{\left[x'(u)^2 + y'(u)^2 + z'(u)^2\right]^{1/2}}{v[x(u), y(u), z(u)]} du. \qquad (1)$$

The signal path between the source *D* and the receiver *R* has the smallest signal travel time. Thus, the sought-for route with the smallest robot travel time from *R* to *D* is identical to the signal path from *D* to *R*. The signal path from *D* to *R* can be found by minimizing *T* with respect to the path function *P*(*u*), that is, by solving a problem of calculus of variations. The problem can be solved in closed form only for simple speed functions *v*(*x*,*y*,*z*); in the general case, numerical methods must be used.

For the present application, we take an indirect approach to the numerical solution. Assume that the function *H*(*x*,*y*,*z*) of signal arrival times from *D* are known. Then, the path with the shortest travel time from any initial position *R* to *D* is the steepest descent path in the function *H*. Finding the steepest descent paths by numerical algorithms is simple. Therefore, in this approach, the main numerical problem is finding the signal arrival function *H*(*x*,*y*,*z*) which is available in closed form only for simple speed functions *v*(*x*,*y*,*z*) and simple geometries. To compute *H*(*x*,*y*,*z*), in the general case we propose an algorithm that uses Huygens' Principle in optics, which states that the wave front of a signal is also the envelope of secondary waves emanating from all points of the field.

In our application, the navigation space (navigation terrain) is the *x*,*y*-plane, and the signal arrival function *H*(*x*,*y*) is computed at discrete points of the plane. Specifically, the navigation space is granulated by a square grid and *H* is computed at the centers of the grid cells. The concept of the computation by Huygens' Principle is as follows: we start at *D*, compute the signal arrival times at the centers of cells located next to *D*, and repeat the procedure by starting from the neighbor points until the space of interest is covered. However, to accurately calculate the signal arrival times at neighbor points, one must know the paths of signals between the source and the neighbor points. To find these paths, we must again solve a problem of calculus of variations, but on a smaller scale (from one cell center to the center of a neighbor cell). A practical solution to this problem is obtained by making three simplifying assumptions.

First, we use average navigation speeds instead of the local speed values. That is, to each terrain cell with center coordinates (x,y), we assign an average speed of $v(x,y)$ for that cell. Second, we granulate the direction of the signal propagation by assuming that a signal from one cell center can travel only to the cell centers of its eight neighbor cells. Third, we assume that signals travel along straight lines from one cell center to the next with constant speeds in each cell that equal the average speeds in the corresponding cells.

The use of average speeds is not very restrictive for the modeling of signal propagation because any speed function $v(x,y)$ can be approximated to any desired degree by choosing an appropriate cell size. The granulation of signal propagation directions is more problematic because it does not improve with grid refinement. It has, however, a common sense appeal when dealing with average speeds and the granulation significantly simplifies the algorithm for $H(x,y)$.

The described algorithm (where the signal arrival times are computed for successive cells starting with the source cell D) produces the final signal arrival times for all cells in one sweep if the navigation speed $v(x,y)$ is constant. If it is not constant, then several iterations (repeated sweeps) will generally be needed to arrive at final values of $H(x,y)$. The method as described is, however, not well suited for iterations because it is not self-correcting. A numerical error (such as a rounding error) that produces a value too small for H at any one location, propagates through the field during subsequent sweeps. Therefore, we modify the algorithm and use Huygens' Principle in reverse. Instead of considering every cell as source and computing signal arrival times at its neighbors, we consider every cell as a sink and all its neighbors as sources. That is, we compute the signal arrival times from all neighbors of the reference cell and choose the smallest arrival time for the reference cell. (At the beginning of the calculation, the destination cell D is assigned zero signal arrival time, which is not updated. All other cells receive an initial value H_{large} that is sufficiently large to also serve as an upper bound for signal arrival times.) This process is self-correcting; any arithmetical errors are eliminated or reduced in subsequent sweeps. Because Huygens' Principle is used for the calculation of the signal arrival function, we call the function $H(x,y)$ a Huygens' relief.

2.2 Vicinity Map Method. In the vicinity-map method, we calculate $H(x,y)$ only in a vicinity map that represents a moderate neighborhood of the robot. If the destination is also in the vicinity map, then H can be calculated as described in section 2.1, and information about the terrain outside the vicinity map is not needed. If the destination is outside the vicinity map, then we assume that the outside terrain is uniform with constant navigation speed that equals an average speed for the area covered by the vicinity map. For such a terrain, Huygens' relief values can be computed in closed form and the closed form formulas are used to compute initial H-values for the boundary cells of the vicinity map. Next, $H(x,y)$ is computed for interior cells of the vicinity map as described in section 2.1, except that now the boundary cells have predetermined initial values that are less than H_{large}.

After Huygens' relief has been established for the vicinity map, the robot's route can be calculated by a local algorithm until the route reaches either the destination or a boundary zone of the map. We define the boundary zone by a distance to the map boundary that is less than the footprint of the robot (a user-defined size of the robot) and the sensing radius of the robot. For robots with no footprint (e.g., when the actual size of the robot is less than a terrain cell) and no sensing devices, we use a boundary zone with a width of two cells. When the computed path reaches the boundary zone, a new vicinity map is prepared which is centered around the last computed position and the process is restarted by computing the Huygens' relief for the new vicinity map.

The initial values of H in the boundary cells represent signal arrival times from sources outside the vicinity map. We assume that these signals travel with constant speed and along straight lines from the sources to the boundary cells. This provides initial boundary values for the vicinity map that have a gradient toward the closest destination; the model is adequate as long as the destination is not very close to the vicinity map (in terms of the cell size). If the destination is close, then the speed in the receiving boundary cell is also taken into account by the following approximate formula. Let D_{ij} be the distance between the cell center of a boundary cell (i,j) and the destination, d_{ij} be the cell size, v_{ij} be the navigation speed in the cell, and v_{out} be the constant speed outside the vicinity map. Then, the initial boundary value of Huygens' relief in the boundary cell is

$$H_{ij} = \left(D_{ij} - d_{ij}/2\right)/v_{out} + d_{ij}/\left(2v_{ij}\right). \tag{2}$$

In cases with several destinations outside the vicinity map, the signal arrival times from all such destinations are computed with equation (2) and the smallest arrival time among the calculated values is chosen as the initial value for the boundary cell.

If all destinations are inside the vicinity map, then there will usually be no need for Huygens' relief outside the map; therefore, initial boundary values do not need to be computed. Exceptions might be cases where the selected map size is too small for the problem so that all destinations are separated from the robot by impenetrable obstacles that extend over the whole width of the vicinity map and all possible routes are partially outside the map. In such cases, the proper approach is to use larger vicinity maps.

After the initial values of the boundary cells are fixed, all other receiving cells within the vicinity map are assigned a value H_{large} that exceeds all possible signal arrival times for the problem at hand. If there are source cells, then they receive the value $H = 0$ that is not updated by subsequent iterations.

Next, the final relief values are calculated iteratively by sweeping over the vicinity map. In this iteration, the value at the center point of each interior non-source cell is determined as the lowest signal arrival time among the arrival times from all eight neighbor cells. The formula for signal arrival time is as follows. Let x_{ij} and y_{ij} be the coordinates of a cell center, v_{ij} be the navigation speed in the cell, and H_{ij} be the signal arrival time. If the navigation speed is finite, then the arrival time of a signal that arrives in the cell (i, j) from a neighbor cell (k, l) is

$$H_{ij}^{kl} = H_{kl} + 0.5\left(1/v_{kl} + 1/v_{ij}\right)\left[\left(x_{kl} - x_{ij}\right)^2 + \left(y_{kl} - y_{ij}\right)^2\right]^{1/2} . \tag{3}$$

If a navigation speed v_{ij} or v_{kl} is zero (meaning impenetrable cells), then the corresponding H_{ij}^{kl} is set equal to H_{large}. The updated arrival time H_{ij} in the cell (i, j) is set equal to the minimum of the values H_{ij}^{kl} over the index sets $i-1 \le k \le i+1$, $j-1 \le k \le j+1$ (that is, over all eight neighbor cells of cell $[i, j]$), and H_{large}. For boundary cells, equation (3) is applied only for neighbors (k,l) that are inside the vicinity map; the minimum is selected among those calculations and the initial boundary value computed with equation (2). The iteration is terminated when all values of H_{ij} have converged. The theoretical maximum number of iterations is equal to the number of cells in the map. However, in practical applications, the number of iterations can be contained to about ten or less by a proper arrangement of sweep directions and sequences. A useful general sweep arrangement is to start with a radial sweep emanating from a source, followed with line sweeps in alternating directions. The first (radial) sweep usually produces the final values in all but a few cells. The subsequent sweeps in different directions take care of signal refractions around low-speed areas.

The computed relief has no relative minima inside the map because each inside cell has at least one neighbor that is a source cell with a lower relief value. Cells that have the value H_{large} after convergence cannot be reached by a signal from any source (destination) cell. Therefore, H_{large} indicates those areas from which no destination can be reached by a robot.

The robot's route within the vicinity map is determined stepwise by proceeding from the robot's position to that neighbor cell which is a source for the signal arrival, according to Huygens' relief. The neighboring source cells are found by recalculating the arrival times with equation (3). If there is more than one source, then a tie-breaking algorithm alternatively chooses the rightmost and leftmost source cell as the next position of the robot. If the robot has no sensing devices, then the next step is calculated by the same algorithm for the new position, and the process is repeated until the destination or a boundary of the map is reached. If the robot is sentient, then the robot's surroundings are scanned from the new position. If new terrain information has been received, a corresponding new Huygens' relief is calculated. The subsequent step is planned based on the new Huygens' relief. Because there are no relative minima within the map, the robot cannot become stuck and its route ends either in a destination, a boundary cell, or (if the goal is terrain exploration) in a cell with $H = H_{large}$.

To apply the described vicinity map method, one must specify the size and resolution of the vicinity map, in addition to the robot's location and the coordinates of the destinations. The choice of the size and resolution depends on the robot's task and on the resolution of the input map. (It is not sensible to use vicinity maps with a finer resolution than the available input map unless the robot is making itself a new map in real time.) One can expect to arrive at different solutions (different routes) for different vicinity map specifications. Examples showing the influence of vicinity map parameters on the solution are discussed in section 4.

2.3 Telescopic Map Method. A variation of the described vicinity map method is the telescopic map method. It also employs a vicinity map with detailed terrain information, but differs from the former method in the treatment of the terrain outside the vicinity map. Instead of assuming that the terrain outside the vicinity map is uniform, a set of coarse maps are used to represent major terrain features outside the vicinity map. The set consists of maps that are centered around the vicinity map, where each larger map is twice the size of its predecessor. The number of cells is the same in all maps—the cells in each larger map are twice as large as the cells in its predecessor (each larger cell contains four smaller cells). To simplify the logic for the computation of the maps, it is assumed that the number of cells in each map is the square of a power of two. (For practical purposes, maps with 32×32, 64×64, or 128×128 cells are most useful.)

One can assume that usually the available (large) input terrain map has the same granulation as the vicinity map. Then, the vicinity map is a simple copy of the terrain map and the surrounding telescopic maps are obtained by averaging over the smaller cells. The number of telescopic maps is chosen such that the penultimate map contains the most distant destination (or the area to be explored, if the goal is area exploration).

The next step is the computation of Huygens' relief. That calculation starts with the largest map and proceeds inward to the smaller maps. The computation in the largest map starts with setting all Huygens' relief values equal to an appropriate upper bound H_{large}. Next, Huygens' relief values in source cells (cells that contain at least one destination or unexplored cell) are set equal to zero if the largest map is the vicinity map, and otherwise equal to

$$H_S = d_{ij} / (2v_{\max}). \tag{4}$$

In equation (4), d_{ij} is the size of the source cell (i, j), and v_{max} is the maximum speed for the present problem (usually corresponding to road speed). H_S is the signal propagation time across one half of the cell size. A positive relief value H_S for source cells in the larger maps is necessary because cells in different maps have different sizes. A relief value of zero models signal propagation with infinite speed across the cell; if the cell (of an outer map) is very large, then the final relief can be distorted. After assigning the proper relief values to the source cells, the values in the remaining cells of the largest map are computed, as described in sections 2.1 and 2.2, by repeated application of equation (3) and without predetermined boundary values. This calculation is done for the whole map, including the central area that is covered by the smaller maps.

When the iteration has converged, the calculated Huygens' relief values in those cells that border the next smaller map are used to calculate initial boundary cell values for the smaller map. The calculation is straightforward, and corresponding formulas are presented in Celmiņš 1999.

The calculation of Huygens' relief in the next smaller map uses the initial boundary cell values as described in section 2.2. After convergence, the initial boundary cell values for the next smallest map are calculated. The process is then repeated for the next smaller map until the relief in the smallest map (the vicinity map) has been obtained. The robot's route in the vicinity map is calculated stepwise as described in section 2.2.

As in the vicinity map method, the specification of the size and resolution of the innermost map is important and should be done according to the needs of the specific task. Examples of the application of the telescopic map method are given in section 4. Numerical experiments with the telescopic map method show that the choice of the vicinity map parameters is less crucial for the telescopic map method because information about distant features of the terrain is taken into account, albeit in rough approximation. On the other hand, the restriction of the number of cells to squares of powers of two can be inconvenient in particular problems.

2.4 Finite Footprints and Sensing Ranges. Some modifications of the route-finding method from the previous section are needed if either the size of the robot is not negligible or if the robot is in exploring mode and has sensing devices with a range that exceeds its size. The size of the robot is important in cases where the terrain contains impenetrable obstacles with small openings. The described route calculation permits routes to pass through any opening if a signal can propagate through it, but a small opening might not be passable by a large robot. Similarly, the path of a finite size robot should not graze impenetrable obstacles, but instead be kept at a half-diameter distance from such walls. These route restrictions can be easily implemented in a navigation function algorithm by inflating all impenetrable obstacles by one half diameter of the robot and by using the modified terrain for the calculation of the navigation function. The inflated obstacles are called C-obstacles (Latombe 1991). In our method, impenetrable obstacles are represented in the terrain map by cells with zero navigation speed. A map with corresponding C-obstacles can be obtained from the original map in one sweep by replacing in reach cell (i, j), the navigation speed v_{ij} with the smallest navigation speed value within the footprint of the robot located at (i, j). This operation not only generates C-obstacles from impenetrable obstacles, but also inflates areas with low navigation speeds. The result is that the planned route of a finite-size robot will be planned at proper distances from low speed areas. The modified speed map is called a C-map; the corresponding Huygens' relief is calculated using the C-map as input. In the vicinity map method, the C-map is produced only for the vicinity map; in the telescopic map method, C-maps are calculated for every map in the set of telescopic maps. The local algorithms for route calculation in Huygens' relief do not need to be changed.

If the sensing range of an exploring robot exceeds its footprint, then the robot is able to observe the (unknown) target area without actually entering it. The visitation of such observation points can be important, for instance, when the target area is located behind a transparent obstacle (river), is inside a building with windows, or is in a wedge between walls. In such cases, the observation points should be treated as destination cells in addition to the exploring cells. A simplistic solution to this problem is to inflate the unknown (exploration) areas by the range of the sensors; however, test calculations with this solution produce routes with undesirable properties. (When the robot is in the sensing neighborhood of the exploration area, the inflated exploration area often includes more than one neighbor cell to the robot's position. The robot then needs criteria for the choice of the next step, but it is not possible to provide reasonable criteria for all situations. Without sophisticated discrimination criteria, the route becomes erratic.) A better solution is to designate as destinations only those observation points from which the target area can be observed but not directly reached. This includes observation points that allow observation through transparent impenetrable obstacles (lakes, windows) or through inflated parts of opaque impenetrable obstacles. Target areas that are not obstructed do not need to be inflated because they will be properly approached without inflation, and the exploration will be completed at the sensor range from the target. For the calculation of Huygens' relief, the observation points that are designated as destinations receive the same fixed value H_S (equation 4) as other destination cells.

3. Exploration Tasks and Algorithms

If the task of the robot is to travel to a destination D, then the calculation of the corresponding Huygens' relief starts with assigning $H = H_S$ for the cell that contains the destination. That cell becomes a source cell for the signal arrival time calculation and acts as an attractor for paths of robots. If the task of the robot is terrain exploration, all areas to be explored are treated as sources; the Huygens' relief values for all cells in those areas are set equal to H_S. The purpose of the exploration can be, for instance, a checking and/or updating of terrain information, mine clearing and/or planting, decontamination, or other activities that require the coverage of an area. The terrain itself (the speed map in our case) might or might not be known at the beginning of an exploration. Determining the exploration path is conceptually similar to finding a path to a predetermined destination in partially known terrain. If the task is to reach a destination, then the robot is given the coordinates of the destination (possibly several destinations). But if the task is exploration, then the robot is given the outline of the area (or areas) that is to be explored. The difference is that in the former case only a few cells are destination cells, but in the latter case, typically many cells are destination cells. The route-finding algorithm from Huygens' relief is the same in either case.

There are, however, certain differences in the implementation of the route-finding algorithms. If the goal is to cover a finite area with observations, then the robot must maintain a map where the unexplored parts are indicated (let the map be called an E-map). At every step (new position) the robot scans its surroundings. If new features are detected, the robot updates the E-map and computes a corresponding Huygens' relief from which the next step is computed, and so on. (For instance, a new feature is the completed scanning of a terrain part, which changes the E-map.) For the relief calculation, the unexplored areas in the E-map are treated as destination areas and are assigned the value of H_S when Huygens' relief is calculated. In addition, $H = H_S$ is also assigned to regions in the vicinity of unexplored areas from where the unknown areas can be observed through impenetrable transparent obstacles. We call a thus modified E-map an S-map. Because the E-map and S-map may be updated at every step, route planning for exploration cannot be done in advance for the whole route (as can be done for navigation in a known terrain to given destinations). Instead, only one step is determined at a time, and determining the next step is postponed until sensor input for the new position is received. In addition, one must now discriminate not only between areas with slow and fast navigation speeds, but also between obstacles that are opaque or transparent for the sensing device. These terrain properties are not necessarily interdependent because a water body can be impenetrable and transparent, and an area with dense vegetation can be penetrable and opaque. The transparency properties of obstacles must be taken into account for the preparation of S-maps.

Describing the algorithm for exploration route-finding, we first consider the case where the area to be explored is located within the vicinity map. Then, the algorithm consists of the following steps:

(1) Obtain an E-map (speed map with indicated unexplored areas).

(2) Prepare a C-map (speed map with inflated obstacles).

(3) Prepare an S-map (with obstacles from the C-map and unexplored areas from the E-map).

(4) Compute Huygens' relief (using the S-map).

(5) Compute the next position of the robot (using the Huygens' relief).

(6) Move the robot to the new position and scan its surroundings.

(7) Check whether the task is complete and whether new information has been obtained.

(8) If the task is not complete and there is no new information, go to step (5).
If the task is not complete and new information has been obtained, go to step (1).

The concepts and properties of C and S-maps are illustrated in Figures 1–4. Figure 1 (an E-map) shows a simple terrain with an impenetrable opaque wall at $y = 4$ m. The wall has a door at $3.2 \text{ m} < x < 5.8 \text{ m}$. There is also a window at $10.2 \text{ m} < x < 12.8 \text{ m}$, which is assumed impenetrable but transparent. The impenetrable wall and window are assigned zero navigation speeds. The navigation speed in the remaining known areas is assumed to be uniform and finite. The two white areas to the south of the wall indicate areas to be explored. For instance, a robot approaching from the north with a sensing radius of 2 m can explore the areas by either passing through the door and turning east, or by remaining to the north of the wall and inspecting the west areas through the door and the east area through the window. (The two curves to the north of the wall indicate two exploration routes of the latter type.)

A Huygens' relief, based only on navigation speed (without taking into account the transparency of the window), is shown in Figure 2; the cell size in this example is $10 \times 10 \text{ cm}^2$. The contour lines in the figure indicate signal arrival times from the unknown areas. The contour lines have corners due to the granulation of signal propagation directions. Because signals can propagate in only eight distinct directions, the contours of signal arrival times from a

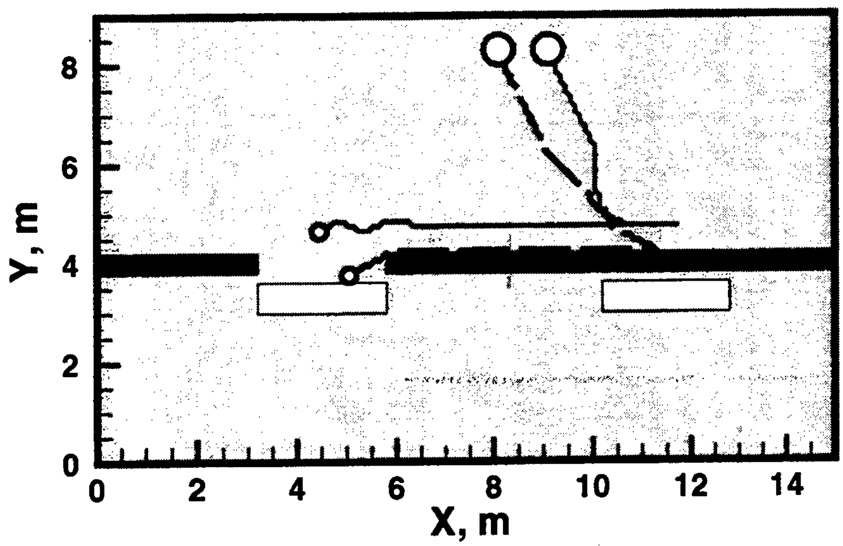

Figure 1. Terrain With Wall.

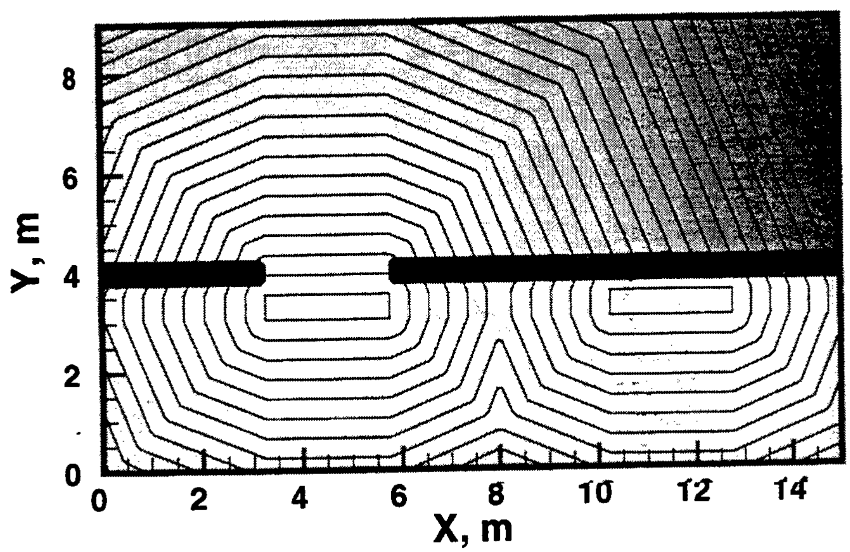

Figure 2. Huygens' Relief for a Solid Wall.

point in a uniform velocity field are octagons instead of circles. It is clear from the figure that for a robot approaching from the north along the steepest descent lines, only the door is an attraction area. The robot will, therefore, pass through the door and continue its exploration to the south of the wall.

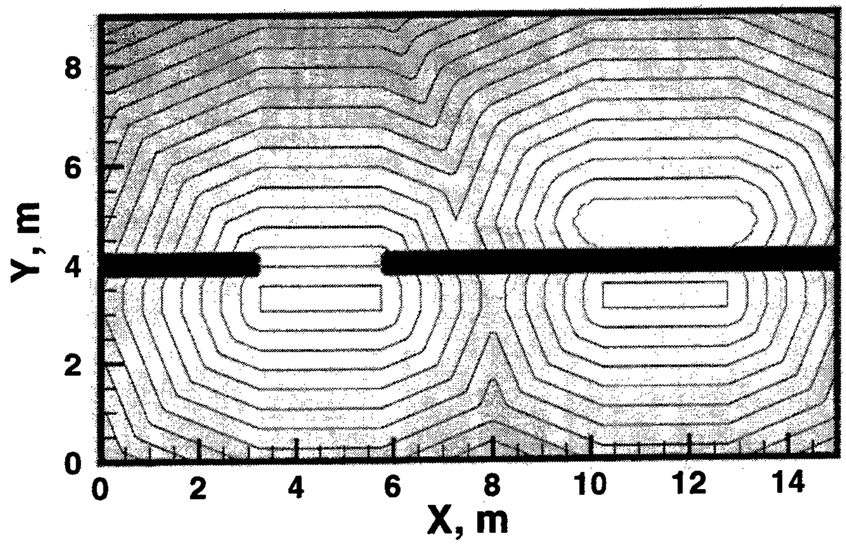

Figure 3. Huygens' Relief for a Wall With Window.

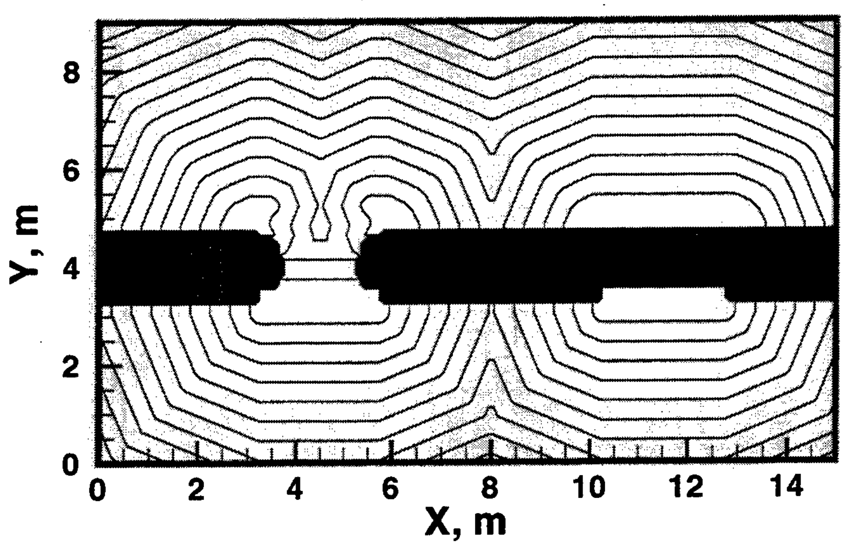

Figure 4. Huygens' Relief for a Robot With Finite Footprint.

Figure 3 shows the Huygens' relief of the same terrain, but based on an S-map corresponding to a robot sensing radius of 2 m. The S-map contains a false unexplored area to the north of the window, shown as a white area in the Huygens' relief. That area is the locus of observation points from where the robot can inspect the unexplored area to the south through the window. The false unexplored area acts as an additional attraction point for route calculation. Using the relief in Figure 3, an exploring robot's route depends on the robot's initial position, which can be either entirely to the north of the wall or can include a pass through the door. Figure 1 shows two possible exploring routes. The thick dashed curve that grazes the wall is the route for an exploring robot with a negligible footprint. The initial part of the route is based on the Huygens' relief that is shown in Figure 3. (Because the relief is recalculated when new observations by the robot are incorporated into the S-map, the latter parts of the route are computed using new, updated Huygens' reliefs.)

For robots with finite footprints, Huygens' relief must be calculated using an S-map (a map with indicated search areas) that is based on a terrain map with inflated obstacles (a C-map). Figure 4 shows the Huygens' relief based on such an S-map for a robot with a circular footprint of 1-m diameter. It shows that the wall and window thicknesses have been increased by 0.5 m in all directions, which among other things reduces the size of the door opening. In addition to the false unexplored area north of the window, there are two new false unexplored areas north of the door. These areas are the loci from where the robot can observe parts of the actual unexplored area through the inflation zone of the wall. In the present example, such false unexplored areas are unnecessary because the robot's footprint does not prevent passage through the door. However, if the door opening were too small to allow the robot's passage through the door, the inflated walls would completely cover the opening and transform the door into a transparent impenetrable obstacle like a window. In such cases, only a false attraction area to the north would enable the robot to find observation points that allow exploration of the unknown area. Therefore, inflation zones of impenetrable obstacles are treated as windows for the preparation of S-maps. An exploration curve for a robot with 1-m diameter is shown in Figure 1 as a thin line. The beginning of the route is calculated using the Huygens' relief of Figure 4. At all times, the route places the robot at an appropriate distance from the wall.

4. Examples

A first example illustrates the exploration of a room with simple obstacles. Figure 5 shows a map of a room where the dark gray areas denote transparent obstacles (such as tables), and the

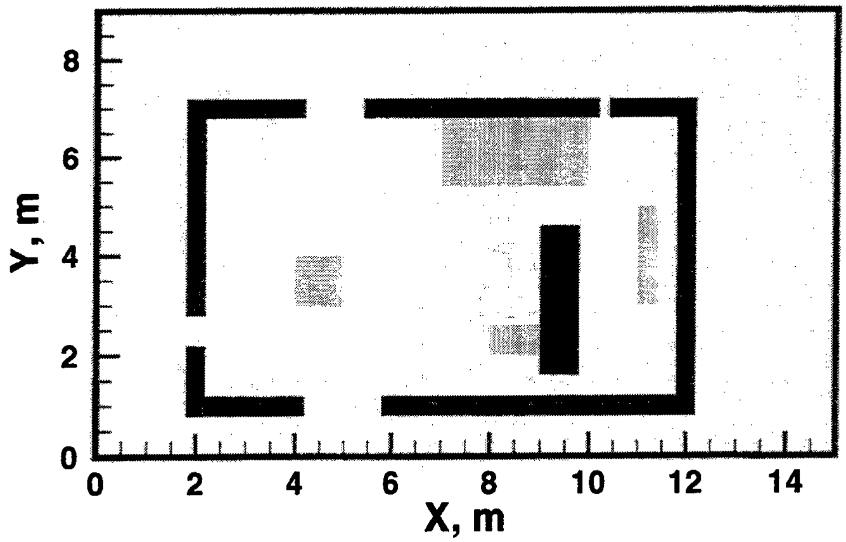

Figure 5. Room With Simple Obstacles.

black areas indicate opaque obstacles (such as walls or closets). We assume that the interior of the room is unknown and to be explored. The granulation in this example is provided by a grid with $20 \times 20\ cm^2$ cells.

Figure 6 shows the Huygens' relief for the room in Figure 5 and for a robot with a negligible footprint. The inside of the room is unknown and treated as source area, indicated in white in this picture. Signals from that source exit through the doors of the room, causing typical relief features that make the door openings attractors for the robot. (The contour lines in this example follow the contours of the computational cells; in section three, interpolated contours were presented).

Figure 7 shows an initial portion of the path of an exploring robot with a 2-m sensing radius and the corresponding partially explored area. The corresponding Huygens' relief is shown in Figure 8.

The completed path and exploration result is shown in Figure 9. The room has been completely mapped, except for the inside of the closet that is not available to the robot.

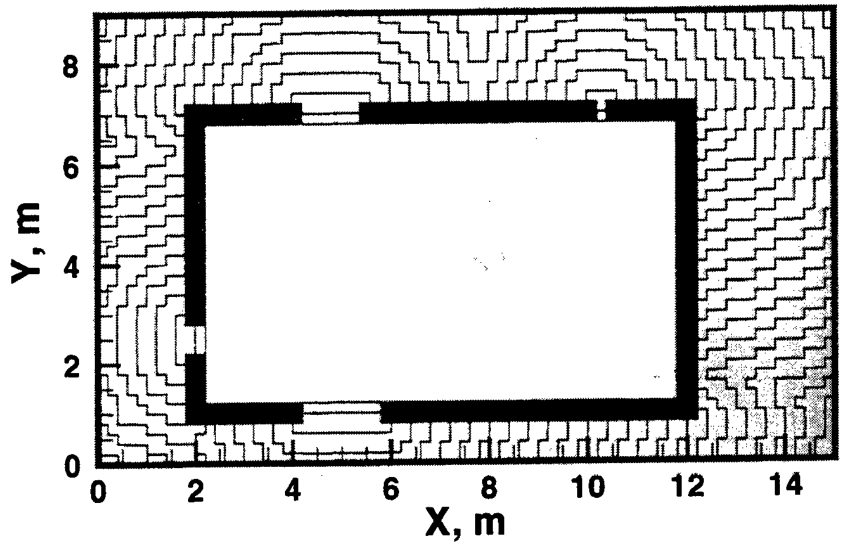

Figure 6. Initial Huygens' Relief for a Room.

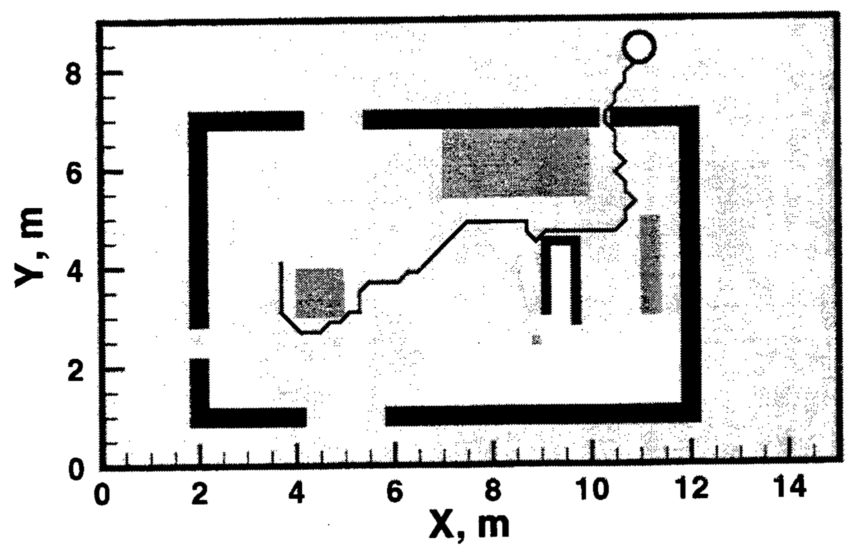

Figure 7. Partially Explored Room.

In this example, we used a map for path finding and navigation that covered the entire room. The use of a single map that covers the entire area of interest is generally recommended for exploration work in maze-like terrain. A vicinity map method or telescopic map method where the vicinity map covers only a fraction of the maze can have problems with finding a reasonable

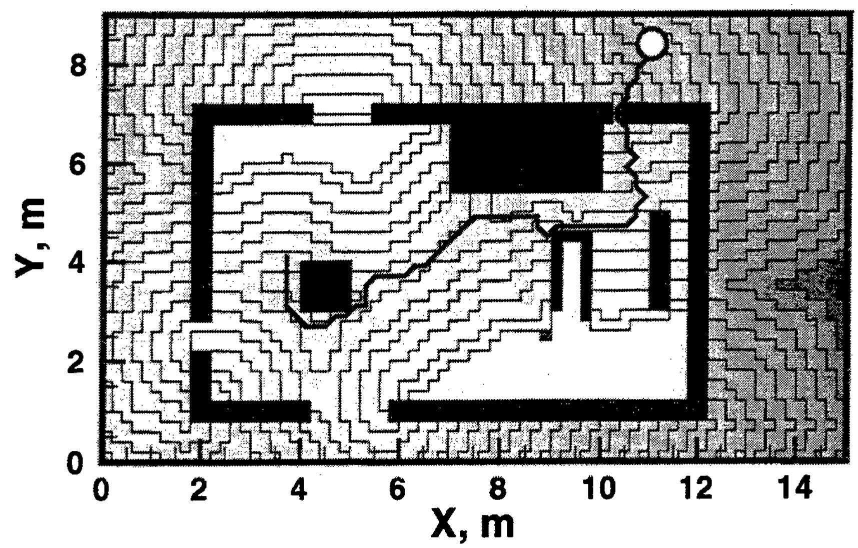

Figure 8. Huygens' Relief for a Partially Explored Room.

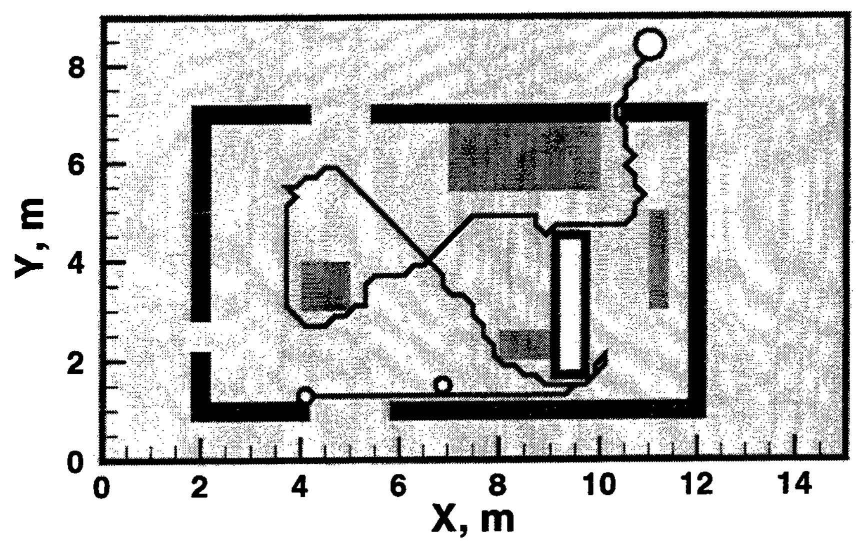

Figure 9. Exploration of a Room by a Small Robot: Complete-Map Method.

route because in these methods, terrain features outside the vicinity map are not fully taken into account. Thus, in the present example, the opaque closet presents a problem when it is not inside the current vicinity map, as illustrated in Figure 10. It shows the exploring path of the same robot as that in Figure 9, but calculated by the vicinity map method using a vicinity map with

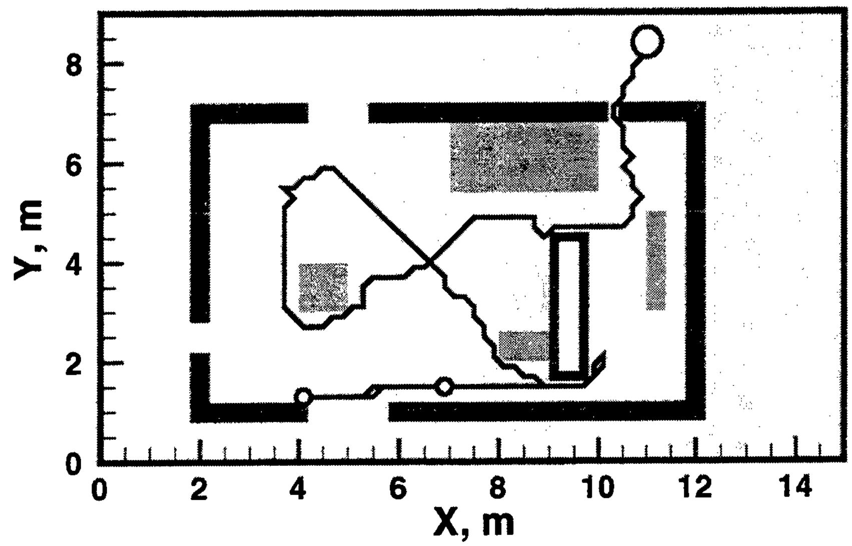

Figure 10. Exploration of a Room by a Small Robot: Vicinity-Map Method.

32×32 cells and a side length of 6.4 m. Most of the exploring path is the same as in Figure 9, but at the very end, the robot reverses itself and travels back to a position with $x \approx 7$ m. This reversal is caused by the unexplored inside of the closet. At the position $x \approx 4$ m (which should be the end position), the closet is outside the current vicinity map and its interior is treated simply as an unexplored outside area. The simplified treatment of outside terrain does not distinguish between accessible and enclosed unexplored areas. Therefore, the robot proceeds towards the closet and the exploration ends only when the entire closet is inside the vicinity map and recognized as an unaccessible unexplored area.

Figures 11 and 12 illustrate the use of the telescopic map method for the same exploration task with the same vicinity map used in the previous example. Figure 11 shows a partially explored room with white areas indicating the unexplored (unknown) parts of the room. The robot's position is indicated by a triangle, and its path up to its present position is shown as a thick curved line. At the indicated position, a new set of telescopic maps have been made with the robot in the center of the vicinity map; the boundaries of the vicinity map and of the next larger map are indicated by heavy straight lines. Figure 12 shows the corresponding simplified terrain map that is used to calculate Huygens' relief. The figure shows that the closet might also

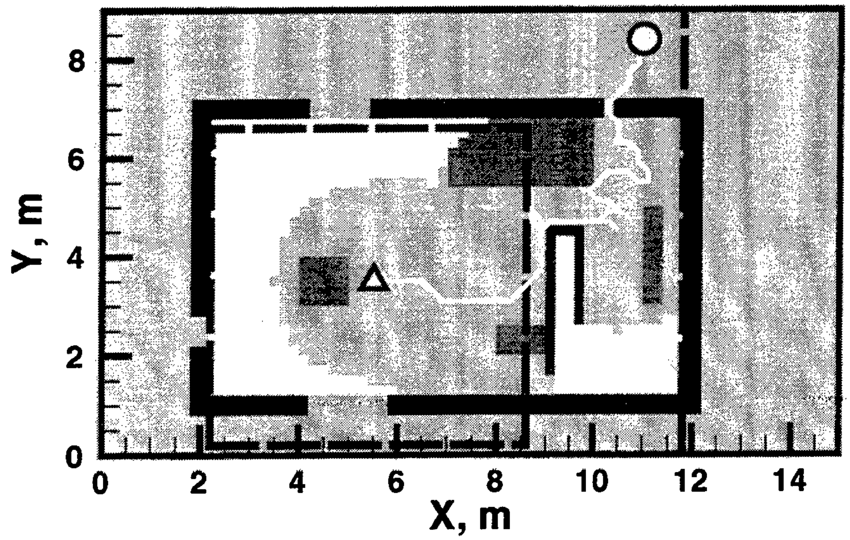

Figure 11. Partially Explored Room: Telescopic Map Method.

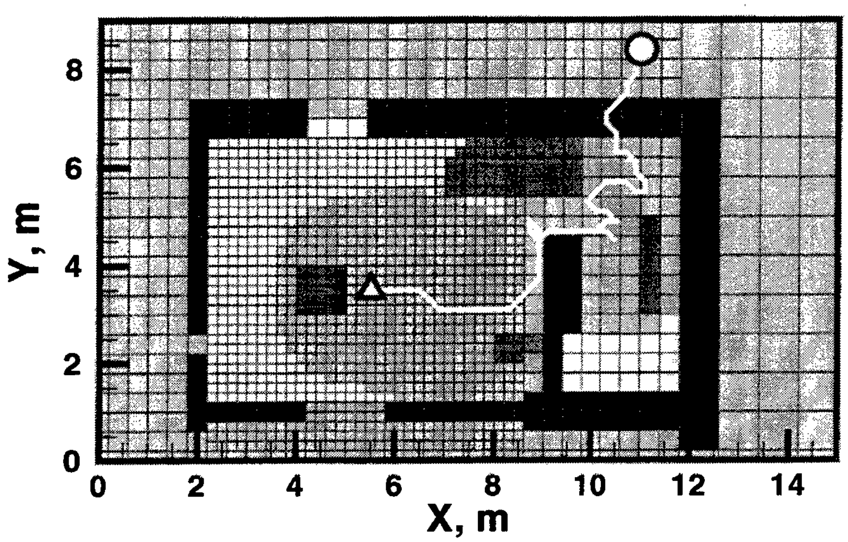

Figure 12. Partially Explored Room in Telescopic Map Representation.

present a problem for the telescopic map method because the 40-cm cells in the larger map are too coarse for an accurate representation of the closet. After exploring the west part of the room, the true nature of the closet will be revealed only when the robot has moved close enough to cover the entire closet by its vicinity map.

In the next example, we specify a robot with a diameter of 0.4 m. A robot with this diameter cannot negotiate the narrow northeast door and is also unable to pass through the space between the closet and the south wall. A corresponding exploring route is shown in Figure 13 and is quite different from that in Figure 9.

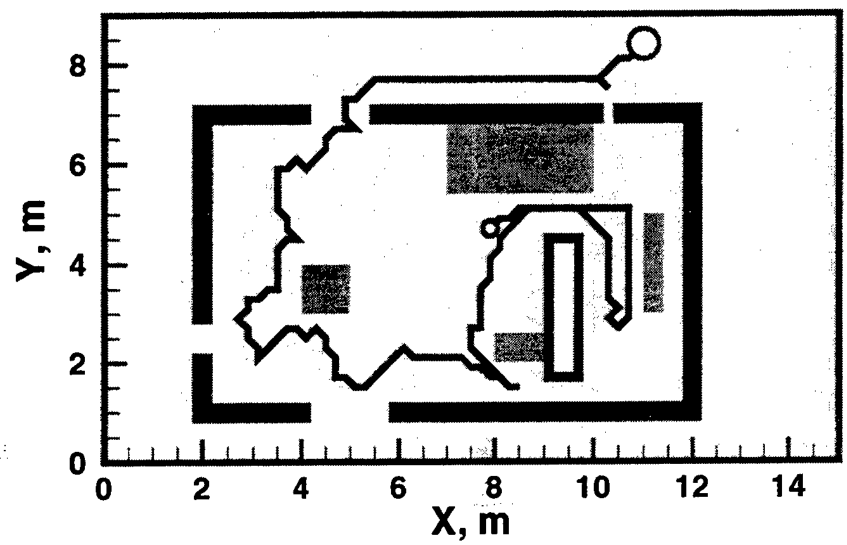

Figure 13. Room Exploration by a Finite Robot.

We now consider navigation in open terrain. An example of a terrain is shown in Figure 14 with the unexplored area indicated by a dashed frame in the center part. Areas with different navigation speeds are indicated by shades of gray. The area contains some roads, three wooded areas in the northern part, and a lake to the south from the center. We assume that the navigation speed in the fields is less than the road speed, and that the speed in the wooded areas is less than in open fields. The lake cannot be crossed by the robot because it is assigned zero speed. There are no impenetrable opaque elements in this scenario.

The initial position of the robot is shown as a large circle in the southwest corner of the map. There are two exploration routes that start from the same initial position. The routes were obtained by the vicinity map method using a vicinity map with a side length of 640 m and $10 \times 10 \text{ m}^2$ cells. The routes correspond to robots with 200-m sensing radiuses but with different footprints. The thick dash-dot curve is the exploration path of a robot with negligible footprint,

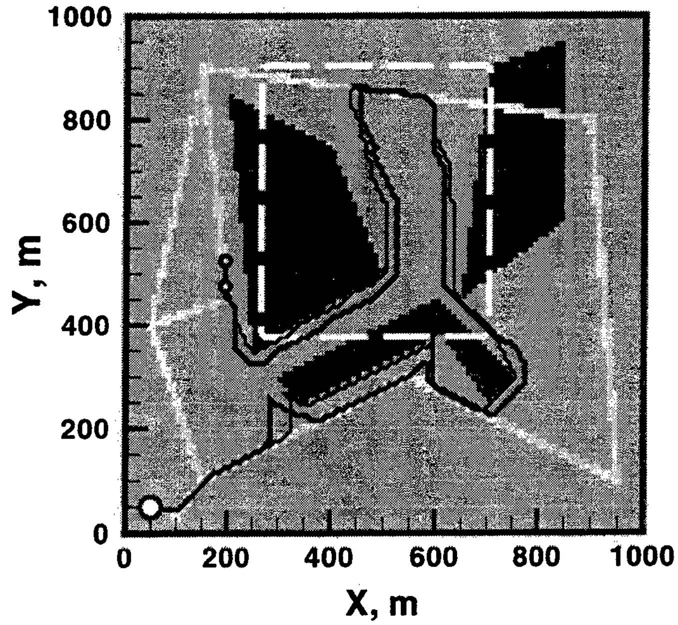

Figure 14. Exploration of a Simple Terrain.

which for the present granulation means a footprint with less than a 10 m diameter. The thinner curve is for a robot with a 50 m footprint. (In the present context, the footprint does not mean the physical size of the robot, but indicates instead that the robot should avoid impenetrable obstacles by a half-diameter of the specified footprint.) A comparison of both paths shows that the robot with the larger footprint does indeed stay away from obstacles and low-speed areas.

To complete the exploration, each robot used a sequence of 20 vicinity maps. Figure 15 shows one of the vicinity maps. The center of the map (the location of the robot at the time the map was generated) is indicated by a triangle. The white area within the wood, to the west of the robot, is yet to be explored; the subsequent path indicates that the exploration is done by navigating along the boundary of the wood. The area outside the vicinity map is assumed to be uniform for the path planning. In this example, the robot needed four more vicinity maps to complete the exploration.

Figure 16 shows the same terrain as it appears in the telescopic map method after partial exploration. The telescopic map method was used with the same cell size as in the previous example ($10 \times 10\,m^2$ cells), but the size of the vicinity map was only half as large. The side length of the vicinity map was 320 m and the diameter of the robot was assumed negligible. One

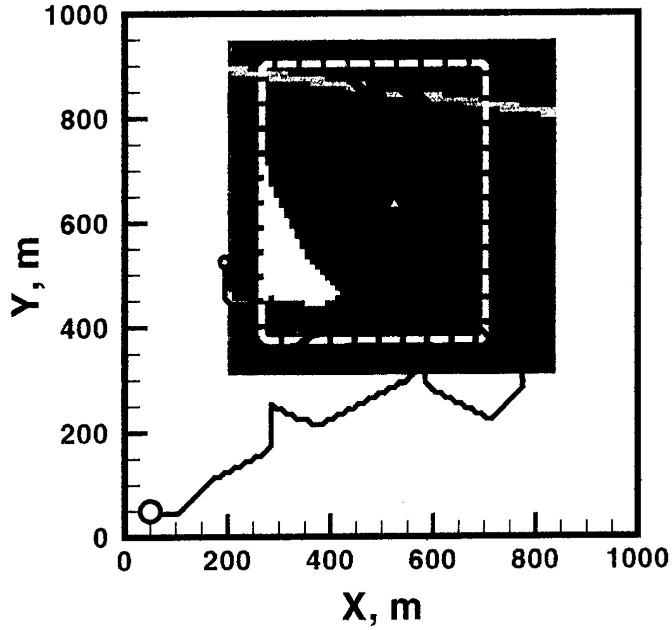

Figure 15. Vicinity Map for Terrain Exploration.

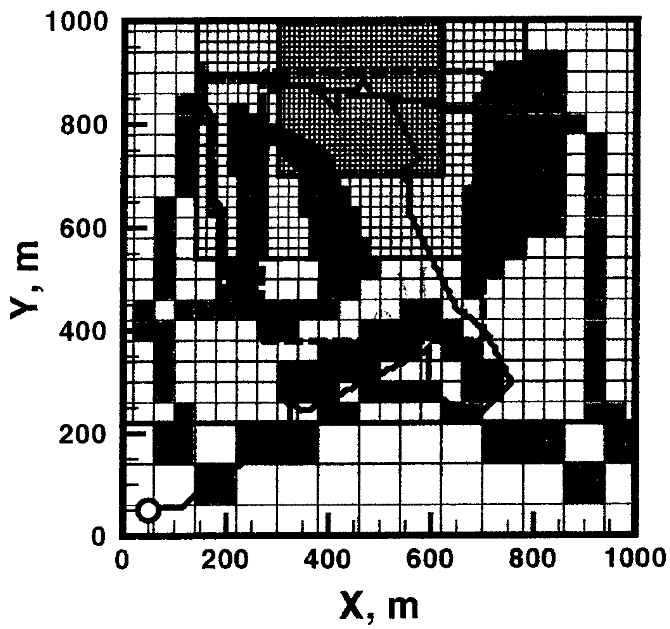

Figure 16. Set of Telescopic Maps for Terrain Exploration.

notices that the exploration paths of the two methods (Figures 14 and 16) are different, which is caused mainly by the different sizes of the vicinity maps. When the telescopic map method was used with the same 64 × 64 cell vicinity map that was used for the vicinity map method, then the computed paths were almost identical. The smaller vicinity map in this example produced a faster exploration route of approximately 14.8 min vs. approximately 18.3 min travel time. Neither of the two methods had problems finding appropriate exploration routes. Complications from finite impenetrable structures are less likely to occur in open terrain situations than in maze-like terrains.

The next example illustrates the exploration of an actual terrain. Figure 17 shows a map with $12.5 \times 12.5\ m^2$ cell granulation of an area in Fort Drum, NY (Champion 1998). The different colors in the map indicate different navigation speeds. We have used five speed categories that roughly correspond to roads, open terrain, lightly wooded areas, water, and impenetrable and opaque wilderness areas. (The wilderness areas are dark blue and can be seen at various places along the river banks.) The area to be explored (and assumed unknown to the robot) is indicated by a white frame.

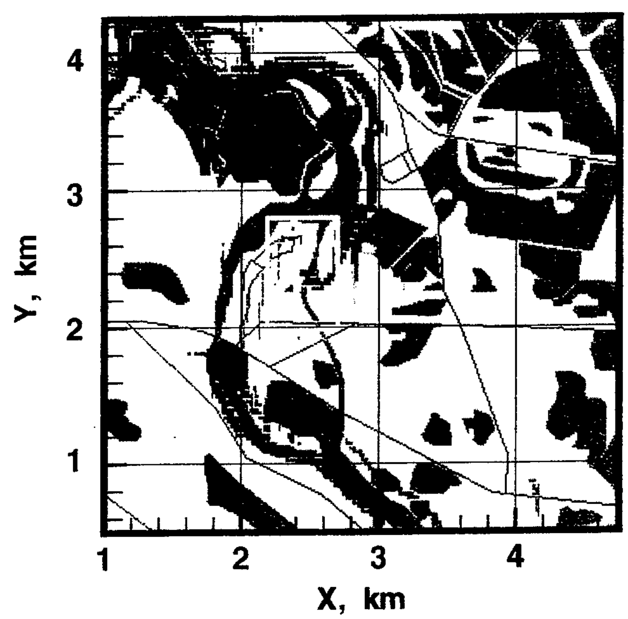

Figure 17. Speed Map of a Fort Drum, NY Area.

Figure 18 shows an exploring robot's path that is obtained with the vicinity map method. The robot has a negligible footprint (less than 12.5-m diameter) and a sensing radius of 200 m. Its initial position is indicated by a large circle in the northeast area of the map. The side length of the vicinity map was 800 m, and the cell size was the same as that in the input map. Hence, the vicinity map had a grid of 64 × 64 cells. A total of 10 vicinity maps were used in sequence to find the route to the unknown area, and another 11 maps were used to explore the area. The figure shows that the robot finds a fast approach to the unknown area by using roads as appropriate, taking a short path through woods, and finding a pass through a wilderness area.

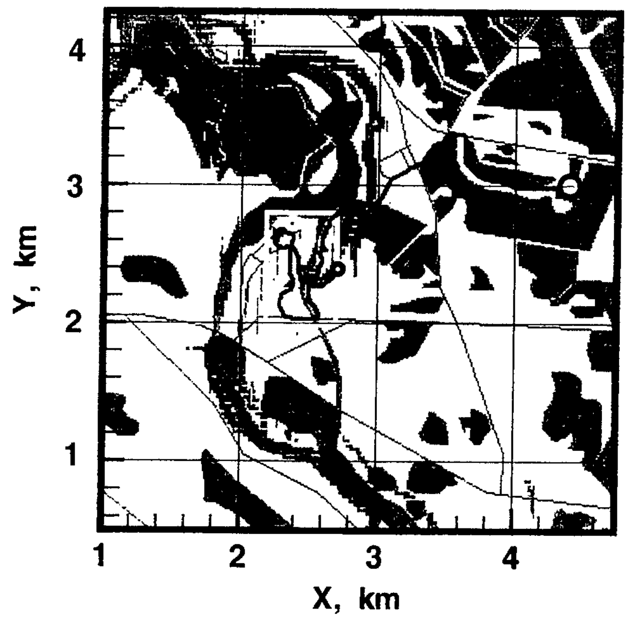

Figure 18. Access Route to Unknown Terrain.

Details of the exploring path within the unknown area are shown in Figure 19. Again, roads and bridges are used where appropriate and obstacles that would slow down the travel are circumnavigated. The final leg of the exploration route is used to explore a small area in the eastern part of the map. That area was shielded by opaque and impenetrable terrain when the robot traveled southward along the river bank at the beginning of the exploration.

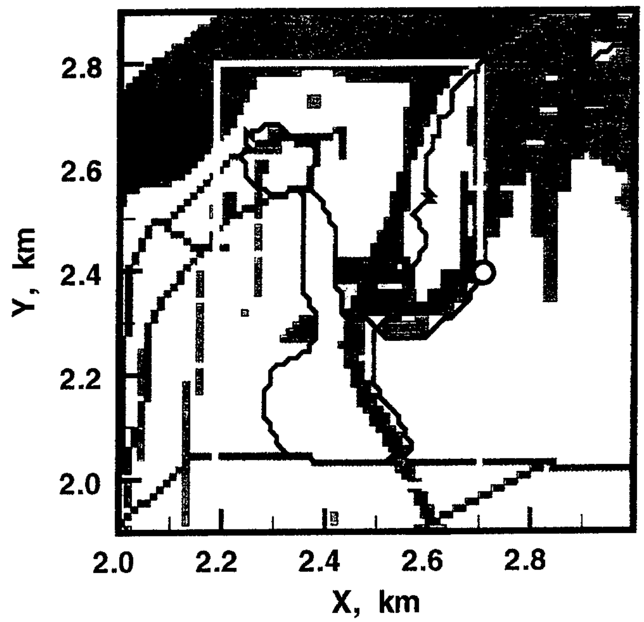

Figure 19. Exploring Route Within Unknown Area.

An example of the situation after a partial exploration is shown in Figure 20. It shows a vicinity map that represents the robot's current world view. The boundaries of the originally unknown area are indicated by the dashed frame. The position of the robot at the time when this map was generated is indicated by a small circle. At that time, unexplored areas are in the northwest corner of the unknown area and behind the opaque obstacle to the east. The dotted line indicates the final leg of the robot's path and shows that the robot will first explore the northwest area and then proceed to the last unknown area to the east. To complete the exploration, four more vicinity maps were used. (Because the sensing radius of the robot was 200 m, a new vicinity map was generated whenever the robot moved to a distance of less than 200 m from a map border.)

We mentioned earlier that the size of the vicinity map should be commensurate with the obstacles in the terrain. On the other hand, a vicinity map that is too large (more than 80 × 80 cells) slows down the route computation. In the example presented, the exploring area and its surroundings contain impenetrable obstacles with some passageways, such as bridges. A proper vicinity map should be large enough to include such passages when they are important for the

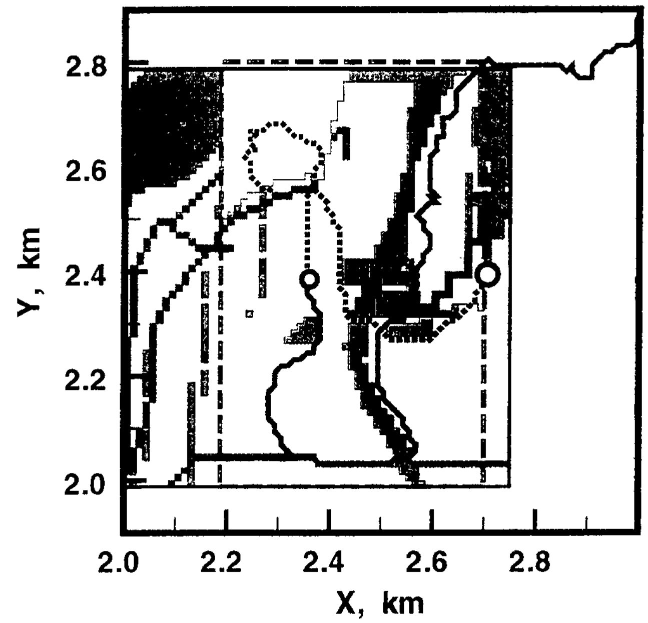

Figure 20. Vicinity Map After Partial Exploration.

approach to the exploring area. For instance, if the task is to find an access to the unknown area from a point in the northwest (across the river), then the vicinity map must be large enough so that the bridge is within the map at the time when the robot arrives at the river's banks. A map that does not contain the bridge cannot be used to find the bridge, and a robot using such a map would become stuck on the west side of the river. (Using the small 800 m × 800 m^2 vicinity map of the example, the robot travels in an infinite loop back and forth along the river bank searching for a passage.) A possible strategy for finding access in this and similar situations is to use a large vicinity map with coarse granulation for the access route and to switch to a smaller map with fine granulation when the exploring area has been reached. Because the robot is frequently updating his internal map, switching to a finer granulation can be easily incorporated into the navigation algorithm. The telescopic map method is a variation of such an approach.

Figure 21 shows the exploration path by the telescopic map method using a vicinity map with 64 × 64 cells, the same as in the previous example. In this case, the telescopic map method produces a different path. Approaching the destination area, the robot uses more roads and circumnavigates the wood to the east of the target area instead of using the short pass through the

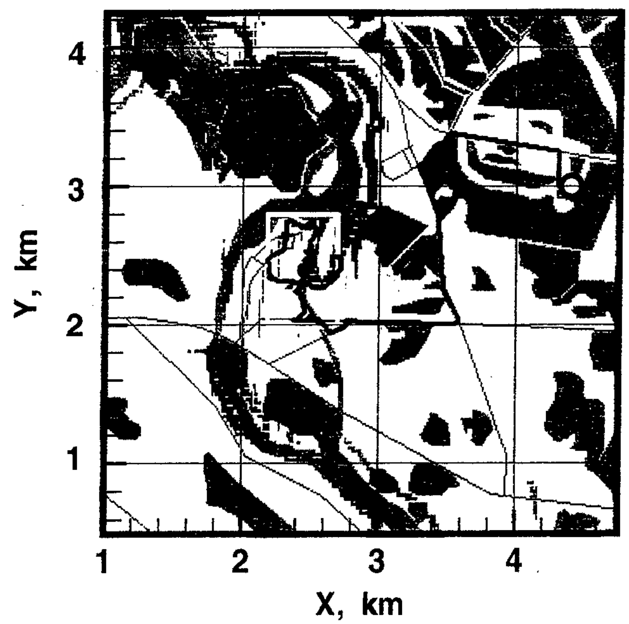

Figure 21. Exploring Route With the Telescopic Map Method.

wilderness. The difference occurs because in the telescopic map method, the terrain outside the vicinity map is represented in the outer (coarser) maps; therefore, the possibility of circumnavigation on roads can be taken into account and compared with the time necessary for crossing the wood and wilderness areas. Inside the unknown terrain, the route is different, of course, because the terrain was entered at a different point. The total travel time for the exploration was 43 min compared to 48 min for the vicinity map method. Hence, the ability to recognize road connections outside the vicinity map has saved some travel time.

5. Summary and Conclusions

In this report, we have described a simple algorithm for determining routes that allow the exploration of specified areas by a robot. The algorithm uses a navigation function that is based on Huygens' Principle in optics and we illustrated how the algorithm can be used for exploration in open terrain with the help of vicinity maps. In this method, the robot keeps a detailed internal map of its neighborhood and determines its next position based on information in the map. As the robot moves, new vicinity maps are established at appropriate intervals. For destinations and exploring areas that are outside the vicinity map, we propose two approaches. In a vicinity map approach, only the coordinates of outside destinations enter the calculation, which provides a general bearing for the route to the destination. In a telescopic map method, the outside terrain is represented by a set of maps that are less detailed at larger distances from the robot. Either of these methods is more efficient than a method that uses large terrain maps at all times. In an open terrain situation, the use of submaps is a necessity.

To use any of the submap methods, one has to specify the resolution and size of the vicinity map. The resolution generally depends on particulars of the exploration task and is usually predetermined. The size of the vicinity map has to be selected by two contradicting requirements: (1) it should be large to cover all salient features of the terrain, and (2) it should be small to save computing time. If important terrain features at larger distances are to be taken into account, then the telescopic map method is likely to provide better results. The computing cost of that method is slightly higher, but it is less sensitive to the choice of the vicinity map parameters.

The route planning algorithms were developed for robot navigation. They can, of course, also be used for route planning in other situations, such as route finding for troops or for any Army units. In the examples in this report we were concerned with finding the fastest routes, but the algorithm is equally applicable to other requirements for the routes.

INTENTIONALLY LEFT BLANK.

6. References

Celmiņš, A. "Battlefield Robot Cybernation and Navigation." ARL-TR-1571, U.S. Army Research Laboratory, Aberdeen Proving Ground, MD, December 1997.

Celmiņš, A. "Route Finding by Huygens' Principle." *Robotics 98: Proceedings of the Third ASCE Specialty Conference on Robotics for Challenging Environments*, Albuquerque, NM, pp. 43–49, April 1998.

Celmiņš, A. "Multimap Procedures for Robot Route Finding in Open Terrain." ARL-TR-1897, U.S. Army Research Laboratory, Aberdeen Proving Ground, MD, February 1999.

Champion, D. Personal communication. TRADOC Analysis Center, White Sands Missile Range, NM, 1998.

Contreras, E. "Remote Area Disassembly Vehicle (RADV) System." *Robotics 98: Proceedings of the Third ASCE Specialty Conference on Robotics for Challenging Environments*, Albuquerque, NM, pp. 113–119, 1998.

Latombe, J. C. *Robot Motion Planning*. Boston: Kluwer Academic Publishers, 1991.

Moore, T. M. "All Purpose Remote Transport System (ARTS) for Active Range Clearance, Force Protection, and Remediation." *Robotics 98: Proceedings of the Third ASCE Specialty Conference on Robotics for Challenging Environments*, Albuquerque, NM, pp. 120–125, April 1998.

Rimon, E., and D. E. Koditschek. "Exact Robot Navigation Using Cost Functions: The Case of Distinct Spherical Boundaries in E^n." *Proceedings of the IEEE International Conference on Robotics and Automation*, Philadelphia, PA, pp. 1791–1796, 1988.

Rimon, E., and D. E. Koditschek. "The Construction of Analytic Diffeomorphism for Exact Robot Navigation." *Proceedings of the IEEE International Conference on Robotics and Automation*, Scottsdale, AZ, pp. 21–26, 1989.

Rimon, E., and D. E. Koditschek. "Exact Robot Navigation in Geometrically Complicated but Topologically Simple Spaces." *Proceedings of the IEEE International Conference on Robotics and Automation*, Cincinnati, OH, pp. 1937–1942, 1990.

INTENTIONALLY LEFT BLANK.

NO. OF COPIES	ORGANIZATION
2	DEFENSE TECHNICAL INFORMATION CENTER DTIC DDA 8725 JOHN J KINGMAN RD STE 0944 FT BELVOIR VA 22060-6218
1	HQDA DAMO FDT 400 ARMY PENTAGON WASHINGTON DC 20310-0460
1	OSD OUSD(A&T)/ODDDR&E(R) R J TREW THE PENTAGON WASHINGTON DC 20301-7100
1	DPTY CG FOR RDA US ARMY MATERIEL CMD AMCRDA 5001 EISENHOWER AVE ALEXANDRIA VA 22333-0001
1	INST FOR ADVNCD TCHNLGY THE UNIV OF TEXAS AT AUSTIN PO BOX 202797 AUSTIN TX 78720-2797
1	DARPA B KASPAR 3701 N FAIRFAX DR ARLINGTON VA 22203-1714
1	NAVAL SURFACE WARFARE CTR CODE B07 J PENNELLA 17320 DAHLGREN RD BLDG 1470 RM 1101 DAHLGREN VA 22448-5100
1	US MILITARY ACADEMY MATH SCI CTR OF EXCELLENCE DEPT OF MATHEMATICAL SCI MADN MATH THAYER HALL WEST POINT NY 10996-1786

NO. OF COPIES	ORGANIZATION
1	DIRECTOR US ARMY RESEARCH LAB AMSRL D D R SMITH 2800 POWDER MILL RD ADELPHI MD 20783-1197
1	DIRECTOR US ARMY RESEARCH LAB AMSRL DD 2800 POWDER MILL RD ADELPHI MD 20783-1197
1	DIRECTOR US ARMY RESEARCH LAB AMSRL CS AS (RECORDS MGMT) 2800 POWDER MILL RD ADELPHI MD 20783-1145
3	DIRECTOR US ARMY RESEARCH LAB AMSRL CI LL 2800 POWDER MILL RD ADELPHI MD 20783-1145
	ABERDEEN PROVING GROUND
4	DIR USARL AMSRL CI LP (BLDG 305)

NO. OF COPIES	ORGANIZATION
1	OUSD AT STTS LW M TOSCANO PENTAGON RM 3B1060 WASHINGTON DC 20301
2	US ARMY TACOM AMSTA ZR B BRENDLE AMSTA TR V P LESCOE WARREN MI 48397-5000
1	US ARMY MICOM UGVS/JPO AMCPM UG COL J KOTORA REDSTONE ARSENAL AL 35802
6	NIST ROBOT SYS DIV BLDG 220 RM B 124 M MURPHY S LEGOWIK M JUBERTS S SZABO H SCOTT DR J ALBUS GAITHERSBURG MD 20899
1	TRADOC ANALYSIS CTR D CHAMPION WSMR NM 88002-5502
1	TOOELE ARMY DEPOT SDSTE AE F ELDREDGE BLDG 1005 TOOELE UT 84074-5004
1	SUPT US MILITARY ACADEMY MADNA A WEST POINT NY 10996-1786
1	US MILITARY ACADEMY DEPT OF MATH SCI CPT R STURDIVANT WEST POINT NY 10996-1786

NO. OF COPIES	ORGANIZATION
1	NCCOSC RDTE DIV 531 B EVERETT 53406 WOODWARD RD SAN DIEGO CA 92152-7383
1	UNIV OF TEXAS ARLINGTON COLLEGE OF ENG DR E G METALLIA PO BOX 19019 ARLINGTON TX 76019-0019
1	STANFORD UNIV COMPUTER SCI DEPT PROF J C LATOMBE STANFORD CA 94305-9010
4	OAK RIDGE NATIONAL LABS J HERNDON DR W HAMEL D HALEY B RICHARDSON PO BOX 2008 OAK RIDGE TN 37831-6304
2	UNIV OF PENNSYLVANIA GRASP LABORATORY PROF R BAJCSY DR E LARGE 34TH AND SPRUCE ST PHILADELPHIA PA 19104
1	USAE WTRWYS EXPRMNT STATION INFORMATION TCHNLGY LAB C D BULLOCK VICKSBURG MI 38180-6199
1	LOCKHEED MARTIN ASTRONAUTICS M S H8380 M ROSENBLUM PO BOX 179 DENVER CO 80201
1	THE ASSOC FOR UNMANNED VEHICLE SYSTEMS INT B LINDAUER 135 N LYNN ST STE 950 ARLINGTON VA 22209-2022

NO. OF COPIES	ORGANIZATION
1	DARPA T M STRAT 3701 N FAIRFAX DR ARLINGTON VA 22203
2	RAND CORP M S DTP 2 R STEEB J M MATSAMURA 1700 MAIN ST PO BOX 2138 SANTA MONICA CA 90407-2138
1	CARNEGIE MELLON UNIVERSITY THE ROBOTICS INSTITUTE DR TERRY R PAYNE 5000 FORBES AVENUE PITTSBURGH PA 15213
1	UNIVERSITY OF PITTSBURGH SCHOOL OF INFORMATION SCIENCES DR. MICHAEL LEWIS 135 N BELLEFIELD AVE PITTSBURGH PA 15260
15	DIRECTOR US ARMY RESEARCH LAB AMSRL CI P EMMERMAN J GANTT COL M KINDL AMSRL CI CN P FISHER D HILLIS T MILLS J VAN ETTEN AMSRL CI CB J BOWDEN S BRAGONIER P DAVID R GREGORY S HO J LOCKER T ROSE S YOUNG 2800 POWDER MILL RD ADELPHI MD 20783-1145

NO. OF COPIES	ORGANIZATION
	ABERDEEN PROVING GROUND
29	DIR USARL AMSRL WM AMSRL WM BB DR J BORNSTEIN T L BROSSEAU G HAAS B T HAUG C L HENRY M D KREGEL I T MCLAUGHLIN R M PHELPS C M SHOEMAKER J J SPANGLER M TRAN T T VONG R VON WAHLDE N M WOLFE AMSRL HR SC T W HADUCH AMSRL IS CI E BAUR A BRODEEN B BROOME A CELMIŅŠ (3 CPS) J DUMER LTC R HAMMELL T HANRATTY R HELFMAN H INGHAM R KASTE AMSRL CI HC C ZOLTANI

INTENTIONALLY LEFT BLANK.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 2000	Final, Jun 98 – Jul 99

4. TITLE AND SUBTITLE
Terrain Exploration by Autonomous Robots

5. FUNDING NUMBERS
611102.H48

6. AUTHOR(S)
Aivars Celmiņš

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
U.S. Army Research Laboratory
ATTN: AMSRL-CI-CT
Aberdeen Proving Ground, MD 21005-5067

8. PERFORMING ORGANIZATION REPORT NUMBER
ARL-TR-2299

9. SPONSORING/MONITORING AGENCY NAMES(S) AND ADDRESS(ES)

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT ***(Maximum 200 words)***

Terrain exploration with the help of autonomous unmanned ground vehicles can be very cost effective in several military applications, such as mine clearing, decontamination, surveillance, and similar operations. A subtask in autonomous operations is planning the traveling route so that the main goal (the exploration of the terrain) is achieved in the shortest time, the entire area of interest is covered, and the terrain features discovered during the exploration are taken into account in real time. In this report, we propose a solution to the subtask. The solution employs a navigation function that is computed using an algorithm based on Huygens' Principle in optics. The proposed solution is an extension of previously reported route-finding methods to predetermined destinations in open terrains. In such terrains, it is necessary to distinguish between terrain in the vicinity of the robot and at farther distances. In the vicinity, the terrain must be carefully examined; therefore, a detailed vicinity terrain map is needed. Terrain features at larger distances can be handled by using approximate representations. In this report, two methods for handling open terrain problems are presented: a vicinity map method and a telescopic map method. The advantages of each method are discussed and examples illustrate how to use the methods in exploration tasks.

14. SUBJECT TERMS
robot navigation, navigation function, terrain exploration, Huygens' Principle, open terrain, vicinity map, telescopic maps

15. NUMBER OF PAGES
40

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18 298-102

INTENTIONALLY LEFT BLANK.

USER EVALUATION SHEET/CHANGE OF ADDRESS

This Laboratory undertakes a continuing effort to improve the quality of the reports it publishes. Your comments/answers to the items/questions below will aid us in our efforts.

1. ARL Report Number/Author ARL-TR-2299 (Celmiņš) Date of Report September 2000

2. Date Report Received ____________________

3. Does this report satisfy a need? (Comment on purpose, related project, or other area of interest for which the report will be used.) ____________________

4. Specifically, how is the report being used? (Information source, design data, procedure, source of ideas, etc.) ____________________

5. Has the information in this report led to any quantitative savings as far as man-hours or dollars saved, operating costs avoided, or efficiencies achieved, etc? If so, please elaborate. ____________________

6. General Comments. What do you think should be changed to improve future reports? (Indicate changes to organization, technical content, format, etc.) ____________________

CURRENT ADDRESS

Organization

Name E-mail Name

Street or P.O. Box No.

City, State, Zip Code

7. If indicating a Change of Address or Address Correction, please provide the Current or Correct address above and the Old or Incorrect address below.

OLD ADDRESS

Organization

Name

Street or P.O. Box No.

City, State, Zip Code

(Remove this sheet, fold as indicated, tape closed, and mail.)

(DO NOT STAPLE)