Submitted to *Autonomous Robots*, Special Issue on Learning in Autonomous Robots.

Integrating Exploration and Localization for Mobile Robots

Brian Yamauchi, Alan Schultz, and William Adams

Navy Center for Applied Research in Artificial Intelligence
Naval Research Laboratory
Washington, DC 20375-5337

{yamauchi, schultz, adams}@aic.nrl.navy.mil

Abstract

Exploration and localization are two of the capabilities necessary for mobile robots to navigate robustly in unknown environments. A robot needs to explore in order to learn the structure of the world, and a robot needs to know its own location in order to make use of its acquired spatial information. However, a problem arises with the integration of exploration and localization. A robot needs to know its own location in order to add new information to its map, but a robot may also need a map to determine its own location. We have addressed this problem with ARIEL, a mobile robot system that combines frontier-based exploration with continuous localization. ARIEL is capable of exploring and mapping an unknown environment while maintaining an accurate estimate of its position at all times. In this paper, we describe frontier-based exploration and continuous localization, and we explain how ARIEL integrates these techniques. Then we show results from experiments performed in the exploration of a real-world office hallway environment. These results demonstrate that maps learned using exploration without localization suffer from substantial dead reckoning errors, while maps learned by ARIEL avoid these errors and can be used for reliable exploration and navigation.

1.0 Introduction

Exploration and localization are two of the fundamental capabilities required for mobile robots to navigate robustly in unknown environments. A mobile robot needs to explore in order to learn the spatial structure of the world, and to incorporate this structure into some form of map. A mobile robot needs to localize itself within the world in order to make use of this map.

However, a problem arises with the integration of exploration and localization. The robot needs to know its own location in order to add new information to the map, but the robot may also need a map to determine its own location. Robots often use dead reckoning to estimate their position without a map, but wheels slip and internal linkages may be imprecise. These errors accumulate over time, and the robot's position estimate becomes increasingly inaccurate. An alternative would be to explicitly provide the robot with its own position using GPS or radio beacons. However, GPS is unreliable in indoor environments, and the use of beacons would require modifying the environment in advance.

Thus, for a robot exploring an unknown environment, a key question is how to build a map while simultaneously using that map to self-localize. We have addressed this question with ARIEL (Autonomous Robot for Integrated Exploration and Localization). ARIEL combines frontier-based exploration [12] with continuous localization [10] in a mobile robot system that is capable of exploring and mapping an unknown environment while maintaining an accurate estimate of its position at all times.

Many researchers have built robots that can navigate from one place to another, but few have built robots that can learn the spatial structure of their environment. Many mobile robots are either purely reactive and have no internal model of the world, or they rely upon a predefined model provided by a human who has mapped the world in advance. Purely reactive robots may be robust in a variety of environments, but they lack the ability to navigate to particular destinations, and such an ability is often necessary for performing useful tasks. Robots that use predefined maps are often quite effective within their particular environments, but they lack the ability to adapt to new environments.

Researchers have approached the problem of spatial learning with two different broad categories of approaches. One approach is to use a spatial representation that explicitly models the spatial structure within the world. Representations of this type include occupancy grids, topological maps, and vector fields, among others. Another approach is to use a form of internal state that implicitly models the world through the dynamics of the robot. Such an internal state could take the form of neural networks, production system rules, fuzzy logic rules, or any other control system that lacks an explicit map of the world.

We choose to have our robot explicitly model the world using occupancy grids. While a number of learning robots have been developed that do not use maps, these robots have been limited to very simple environments, and often to a limited subset of possible destinations within those environments. With an explicit map, a path planning algorithm, and a good low-level capability for reactive obstacle avoidance, a robot can navigate to any accessible region represented by an occupancy grid map. In addition, humans can look at the map and understand the representation. This is useful to determine whether the robot has learned a good map of the world, and also to specify destinations within the robot's learned map.

Considerable work has been done in simulated exploration, but these simulations often view the world as a set of floorplans. The blueprint view of a typical office building presents a structure that seems simple and straightforward—rectangular offices, square conference rooms, straight hallways, and right angles everywhere—but the reality is often quite different. A real mobile robot may have to navigate through rooms cluttered with furniture, where walls may be hidden behind desks and bookshelves.

A few researchers have implemented exploration systems using real robots. These robots have performed well, but only within environments that satisfy certain restrictive assumptions. For example, some systems are limited to environments that can be explored using wall-following [8], while others require that all walls intersect at right angles and that these walls be unobstructed and visible to the robot [11]. Some indoor environments meet these requirements, but many do not.

We have developed an exploration strategy for the complex environments typically found in real office buildings. Our approach is based on the detection of *frontiers*, regions on the border between space known to be open and unexplored space. In this paper, we describe how to detect frontiers in occupancy grids and how to use frontiers to guide exploration. We also describe a method for continuously localizing the robot's position by matching the robot's recent perceptions within the learned occupancy grids. Then we show how we integrated frontier-based exploration with continuous localization in ARIEL. Next we provide results from experiments using a real robot to explore a real-world office hallway environment. Finally, we explain how our work relates to previous research in this area, and we summarize the results of our research.

2.0 Frontier-Based Exploration

The central question in exploration is: *Given what you know about the world, where should you move to gain as much new information as possible?* Initially, you know nothing except what you can see from where you're standing. You want to build a map that describes as much of the world as possible, and you want to build this map as quickly as possible.

The central idea behind frontier-based exploration is: *To gain the most new information about the world, move to the boundary between open space and uncharted territory.*

Frontiers are regions on the boundary between open space and unexplored space. When a robot moves to a frontier, it can see into unexplored space and add the new information to its map. As a result, the mapped territory expands, pushing back the boundary between the known and the unknown. By moving to successive frontiers, the robot can constantly increase its knowledge of the world. We call this strategy *frontier-based exploration*.

If a robot with a perfect map could navigate to a particular point in space, that point is considered *accessible*. All accessible space is contiguous, since a path must exist from the robot's initial position to every accessible point. Every such path will be at least partially in mapped territory, since the space around the robot's initial location is mapped at the start. Every path that is partially in unknown territory will cross a frontier. When the robot navigates to that frontier, it will incorporate more of the space covered by the path into mapped territory. If the robot does not incorporate the entire path at one time, then a new frontier will always exist further along the path, separating the known and unknown segments and providing a new destination for exploration.

In this way, a robot using frontier-based exploration will eventually explore all of the accessible space in the world, assuming perfect sensors and perfect motor control. The real question is how well frontier-based exploration will work using the noisy sensors and imperfect motor control of a real robot in the real world. This is one of the questions that this research is intended to address.

2.1 Evidence Grids and Laser-Limited Sonar

We use evidence grids [9] as our spatial representation. Evidence grids are Cartesian grids containing cells, and each cell stores the probability that the corresponding region in space is occupied. Initially all of the cells are set to the prior probability of occupancy, which is a rough estimate of the overall probability that any given location will be occupied. Evidence grids have the advantage of being able to fuse information from different types of sensors.

We use sonar range sensors in combination with a planar laser rangefinder to build our robot's evidence grid maps. Sonar sensors are cheap and widely available, but specular reflections often significantly degrade their accuracy. When a sonar pulse hits a flat surface at an oblique angle, it may reflect away from the sensor. As a result, either the sensor detects nothing, or it senses objects that, like reflections in a mirror, appear to be much farther away than the nearest surface.

These reflections could cause difficulties for frontier-based exploration, not only due to inaccuracies in the map, but also because specular reflections often appear as large open areas surrounded by unknown territory. As a result, the robot could waste a great deal of time trying to reach non-existent frontiers.

In order to reduce the effect of specular reflections, we have developed a technique we call *laser-limited sonar.* This technique combines the advantages of sonar and laser range sensors. Sonar sensors are effective at determining that large volumes of space are clear of obstacles, but they suffer from specular reflections. Laser range sensors are effective at accurately determining the positions of obstacles. However, the inexpensive, triangulation-based laser rangefinders commonly found on mobile robots are limited to detecting obstacles within a plane.

The standard evidence grid formulation assumes that each sensor reading is independent of every other sensor reading. In reality, this is not the case, and we take advantage of this. We use a laser rangefinder in combination with the sonar sensors, and if the laser returns a range reading less than the sonar reading, we update the evidence grid as if the sonar had returned the range indicated by the laser, in addition to marking the cells actually returned by the laser as occupied.

As a result, evidence grids constructed using laser-limited sonar have far fewer errors due to specular reflections, but are still able to incorporate obstacles detected by the sonar below (or above) the plane of the laser. It is possible that obstacles undetected by the laser may cause sonar specular reflections which will be added to the map. However, in practice, we have found that laser-limited sonar drastically reduces the number of uncorrected specular reflections from walls and other large obstacles, which tend to be the major sources of errors in evidence grids built using sonar.

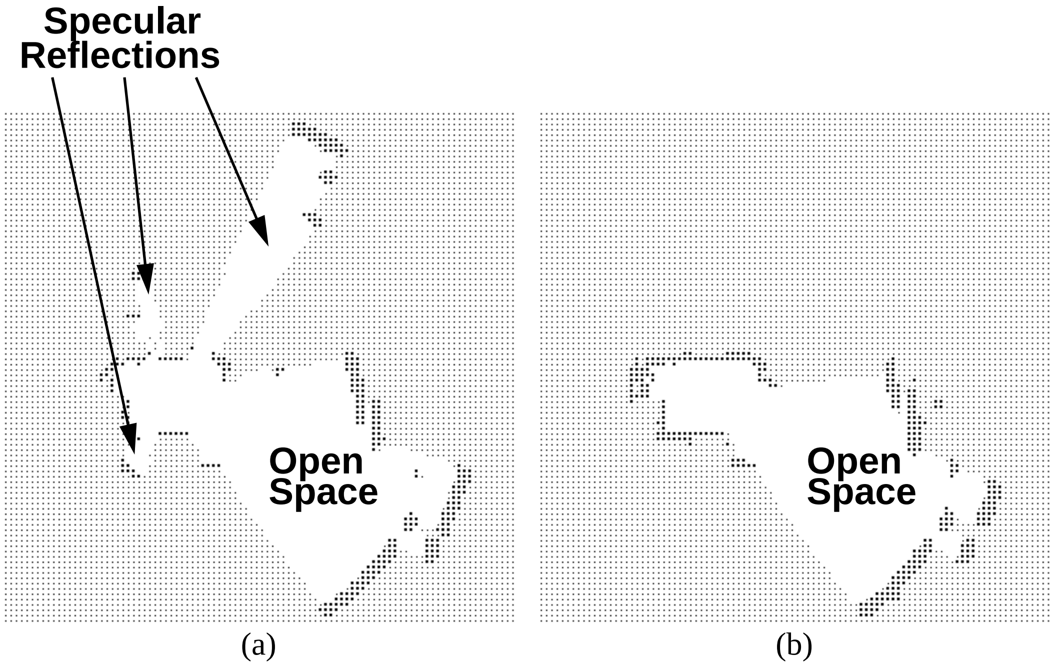

Figure 1: Comparison of evidence grids built using (a) raw sonar and (b) laser-limited sonar

Figure 1 compares the evidence grids constructed using raw sonar (Figure 1a) and laser-limited sonar (Figure 1b). Cells representing open space are represented by whitespace. Cells representing occupied space are represented by black circles. Cells representing unknown territory are represented by small dots.

Both of these grids were built by a robot positioned in an alcove at one corner of a large open area. This open area is visible in the lower right portion of both grids. The grid constructed using raw sonar includes three specular reflections from the adjacent walls. Using laser-limited sonar, these reflections are eliminated.

With a narrow field-of-view laser rangefinder, such as the one currently mounted on our robot, it is necessary to sweep the robot's sensors to obtain a complete picture of the robot's surroundings. With wide field-of-view laser rangefinders, such as the ones we plan to install on our robot in the near future, the robot does not need to stop and sweep its sensors. Instead, laser-limited sonar can be used to build evidence grids while the robot remains in motion.

2.2 Frontier Detection

After an evidence grid has been constructed, each cell in the grid is classified by comparing its occupancy probability to the initial (prior) probability assigned to all cells. This algorithm is not particularly sensitive to the specific value of this prior probability. (A value of 0.5 was used in all of the experiments described in this paper.)

Each cell is placed into one of three classes:

open: occupancy probability < prior probability

unknown: occupancy probability = prior probability

occupied: occupancy probability > prior probability

A process analogous to edge detection and region extraction in computer vision is used to find the boundaries between open space and unknown space. Any open cell adjacent to an unknown cell is labeled a frontier edge cell. Adjacent edge cells are grouped into frontier regions. Any frontier region above a certain minimum size (roughly the size of the robot) is considered a frontier.

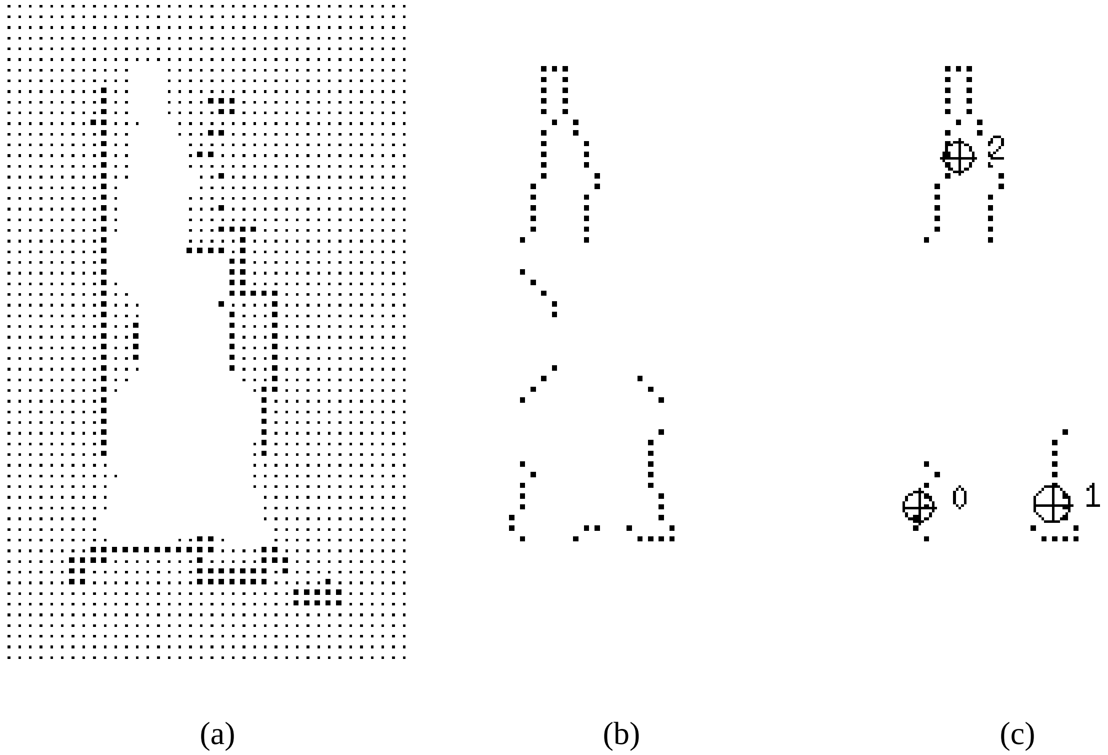

Figure 2: Frontier detection: (a) evidence grid, (b) frontier edge segments, (c) frontier regions

Figure 2a shows an evidence grid built by a real robot in a hallway adjacent to two open doors. Figure 2b shows the frontier edge segments detected in the grid. Figure 2c shows the regions that are larger than the minimum frontier size. The centroid of each region is marked by crosshairs. Frontier 0 and frontier 1 correspond to open doorways, while frontier 2 is the unexplored hallway.

2.3 Navigating to Frontiers

Once frontiers have been detected within a particular evidence grid, the robot attempts to navigate to the nearest accessible, unvisited frontier. The path planner uses a depth-first search on the grid, starting at the robot's current cell and attempting to take the shortest obstacle-free path to the cell containing the goal location.

While the robot moves toward its destination, reactive obstacle avoidance behaviors prevent collisions with any obstacles not present while the evidence grid was constructed. These behaviors allow the robot to steer around these obstacles and, as long as the world has not changed too drastically, return to follow its path to the destination.

When the robot reaches its destination, that location is added to the list of previously visited frontiers. The robot performs a 360 degree sensor sweep using laser-limited sonar and adds the new information to the evidence grid. Then the robot detects frontiers present in the updated grid and attempts to navigate to the nearest accessible, unvisited frontier.

If the robot is unable to make progress toward its destination for a certain amount of time, then the robot will determine that the destination in inaccessible, and its location will be added to the list of inaccessible frontiers. The robot will then conduct another sensor sweep, update the evidence grid, and attempt to navigate to the closest remaining accessible, unvisited frontier.

3.0 Continuous Localization

Without some way to correct for accumulated odometry error, the maps constructed during exploration would become increasingly inaccurate. ARIEL uses continuous localization to compensate for odometry error and maintain an accurate position estimate at all times.

Previous techniques for localization have looked at learning and recognizing landmarks in the environment, either as geometric representations or as a representation of sensor readings. Our localization technique does not rely on the presence of specific landmarks, but instead uses the entire local environment of the robot to determine its location.

An important issue in localization is how often to relocalize the robot in its environment. Many existing techniques only relocalize when either an error in position is detected or after an unacceptable amount of error has accumulated. In continuous localization, ARIEL continuously relocalizes by making regular small corrections instead of occasional large corrections. The benefit is that the error is known to be small, and fast correction techniques can be used.

Continuous localization builds short-term perception maps of its local environment. These maps are of a short duration, and typically contain only very small amounts of positional or rotational error. These short term maps are then used to position the robot within the global map via a registration process, the offset of which is used to correct the robot's current odometry.

In the experiments described in this paper, continuous localization updates the robot's position estimate whenever the robot moves move than 24 inches from the position of its last update. In addition, each degree of (cumulative) rotation is treated as equivalent to 0.03 inches of translation.

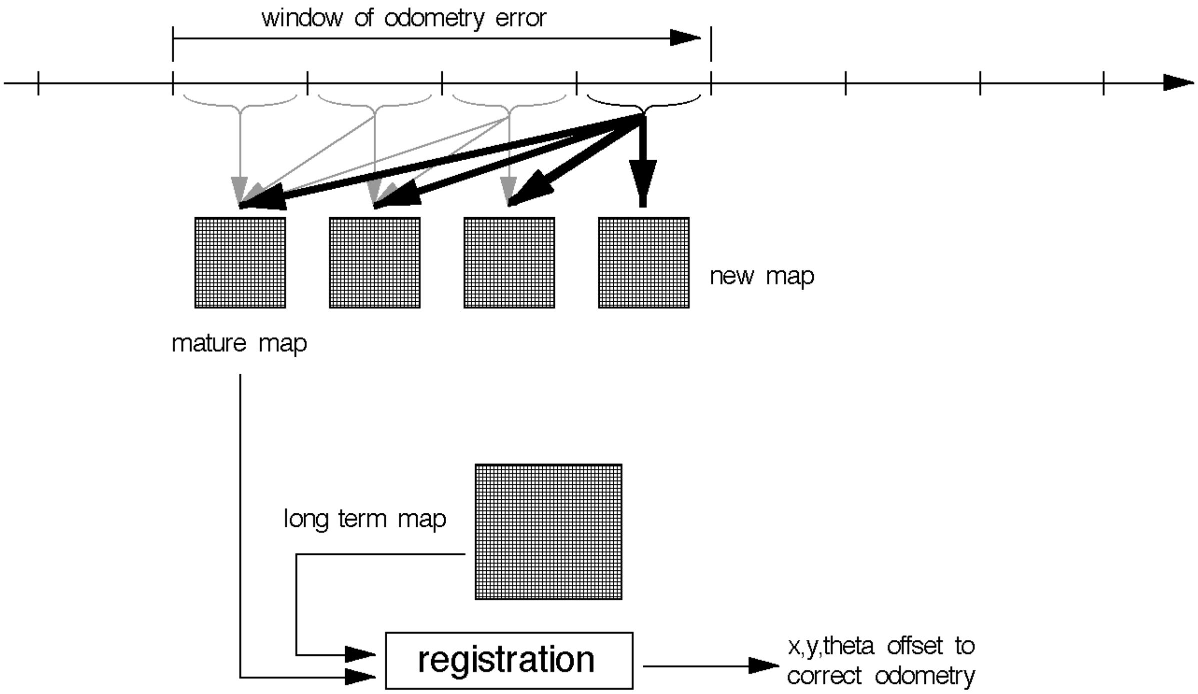

Figure 3: Continuous localization

Figure 3 shows the process of continuous localization. The robot builds a series of short-term perception maps of its immediate environment, each of which is of brief duration and typically contains only a small amount of dead reckoning error. After several time intervals, the oldest (most “mature”) short-term map is used to position the robot within the long-term map via a registration process.

The registration process consists of sampling the possible poses within a small area around the robot’s current pose. For each tested pose, the mature short-term map is adjusted by the difference in pose (the offset) and a match score calculated based on agreement between the cell values of the short-term map and the long-term map, summed across all cells. The match scores for all tested poses are then treated as masses and the offsets as distances, and a center of mass calculation is performed to determine the offset that is likely to have the highest match score. This offset is applied to the robot’s odometry, placing it at the pose which causes its local perceptions to best match the long-term map. All subsequent robot processes use this new odometry. After the registration takes place the most mature map is discarded.

In this previous research with a fixed map [10], we have shown that continuous localization is capable of eliminating accumulated odometry errors with a resulting constant translational error on the order of five inches, or approximately the size of an evidence grid cell. Recent research has also shown that by updating the global map using information from more recent short-term maps, continuous localization can adapt to changing environments [4].

However, in our previous work, continuous localization relied upon an a priori map built by manually positioning the robot at locations throughout the environment and sweeping the sensors. In contrast, ARIEL uses frontier-based exploration to construct the long-term maps used by continuous localization.

4.0 ARIEL: Autonomous Robot for Integrated Exploration and Localization

4.1 System Overview

Frontier-based exploration provides a way to explore and map an unknown environment, given that a robot knows its own location at all times. Continuous localization provides a way for a robot to maintain an accurate estimate of its own position, as long as the environment is mapped in advance. The question of how to combine exploration with localization raises a "chicken-and-egg" problem: the robot needs to know its position in order to build a map, and the robot needs a map in order to determine its position.

ARIEL is designed to address this problem. We assume that the robot starts with an accurate initial position estimate, so localization only needs to correct for dead reckoning errors that accumulate while the robot moves through the world. However, these errors can accumulate quickly, so it would not be feasible to map a large environment using dead reckoning alone.

The solution is to use the partial maps constructed by frontier-based exploration These maps are incrementally extended whenever the robot arrives at a new frontier and sweeps its sensors. Even though these maps are incomplete, they describe the spatial structure of the robot's immediate environment, including all of the territory between the robot's current location and all of the detected frontiers. These maps are passed to continuous localization to be used as long-term maps.

As the robot navigates to the next frontier, continuous localization constructs short-term maps that represent the robot's recent perceptions. If dead reckoning error starts to accumulate, these short-term maps will deviate from the long-term map. The registration process will then correct for this error by adjusting the robot's position estimate.

When the robot arrives at the new frontier, its position estimate will be accurate. When frontier-based exploration performs the next sensor sweep, the new information will be integrated at the correct location within the map.

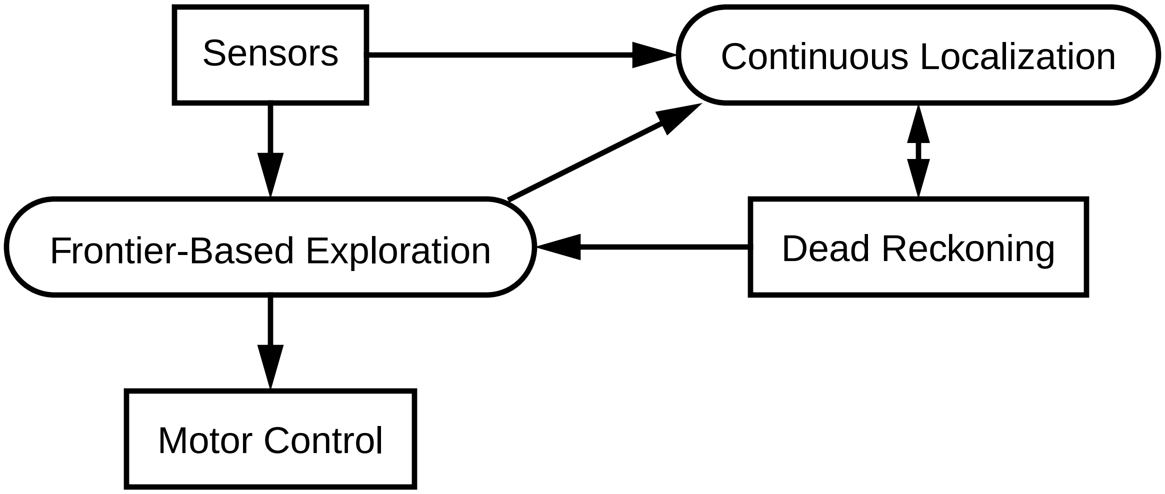

Figure 4: ARIEL system architecture

Figure 4 shows the system architecture for ARIEL. Frontier-based exploration and continuous localization run in parallel. Both processes make use of information from the robot's sensors, but only frontier-based exploration sends commands to the robot's motor control system. Frontier-based exploration passes a new map to continuous localization every time the robot arrives at a new frontier. Continuous localization corrects the robot's dead reckoning transparently, so no direct communication is necessary from localization to exploration.

Frontier-based exploration actively directs the robot's sensing while continuous localization passively collects sensor information that the robot acquires. Reactive obstacle avoidance (within frontier-based exploration) makes use of all sixteen sonar and infrared sensors. The actual mapping process makes use of only the single, laser-limited sonar sensor that is swept at frontiers to obtain a complete picture.

Continuous localization uses data from the laser-limited sonar to build its short-term maps, but only when this sensor is *not* being swept at frontiers. The reason for this is that the sensor data collected at the frontier will immediately be introduced into the long-term map that is passed to continuous localization. If identical data were also placed in the short-term map, it would cause localization to undercorrect for future errors in dead reckoning.

4.2 Implementation

ARIEL is implemented on a Nomad 200 mobile robot is equipped with a planar laser rangefinder, sixteen sonar sensors, and sixteen infrared sensors. Frontier-based exploration and continuous localization run on separate Sparcstation 20s that communicate with each other over an ethernet and with the robot over a radio ethernet. A Pentium processor onboard the robot handles low-level sensor processing and motor control.

When using a narrow field-of-view laser rangefinder, such as the one on the Nomad 200, we have found it useful to aim the sensor backwards as the robot navigates to the next frontier. This allows continuous localization to observe previously-visited territory with the laser-limited sonar. The advantage is that this territory is within the space already mapped by frontier-based exploration in the long-term map. As a result, the short-term maps constructed by continuous localization will have a large degree of overlap with the long-term map, and will provide more accurate registration. If the laser-limited sonar is aimed forward, then as the robot moves to the new frontier, this sensor will return information about space that has not been visited. Then the short-term map will have less overlap with the long-term map, and the quality of the registration will be reduced.

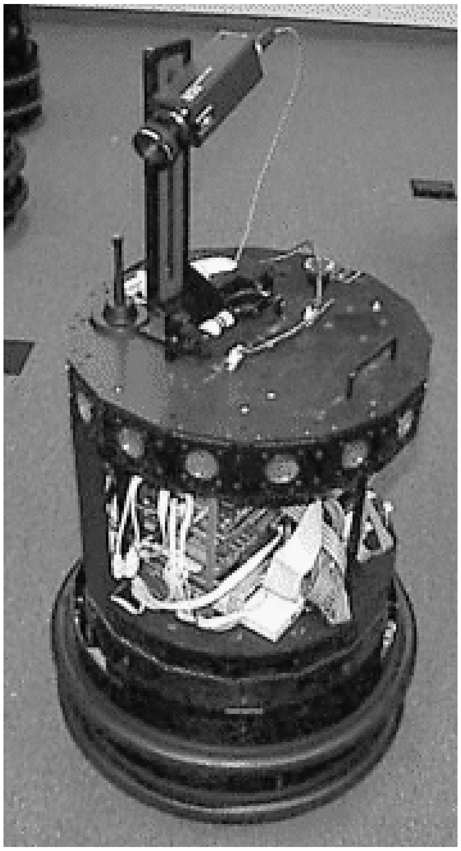

Figure 5: Nomad 200 mobile robot

5.0 Experiments

5.1 Overview

In previous work [12], we have demonstrated that frontier-based exploration can successfully map real-world office environments cluttered with a variety of obstacles: chairs, desks, tables, bookcases, filing cabinets, sofas, water coolers, and boxes of varying size and shape. In relatively small environments, such as a single office or laboratory, frontier-based exploration was capable of mapping accurately without continuous localization. However, for larger environments, such as those containing long hallways, large amounts of position error can accumulate within dead reckoning, and localization is necessary for building accurate maps.

For this reason, we have conducted a set of experiments in a hallway environment (70 feet long). This hallway, like many of those in office buildings, is cluttered with obstacles. These obstacles include a printer table that blocks half the width of the hallway, a set of open cabinets containing electrical wiring, switchboxes mounted on the walls, various cardboard boxes, a water fountain, and a water cooler.

In order to measure ARIEL's performance, we initially constructed a ground truth grid by manually positioning the robot at viewpoints throughout the hallway and sweeping the robot's sensors. This ground truth grid is only used to score the grids learned by ARIEL. The ground truth grid is ***not*** used by ARIEL for exploration or localization.

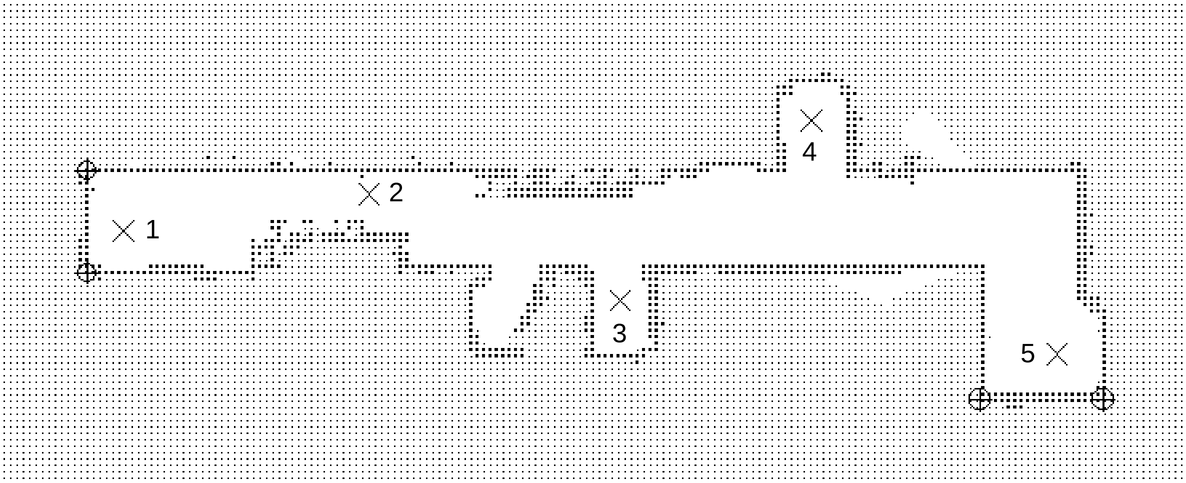

Figure 6: Ground truth evidence grid for hallway

Figure 6 shows the ground truth evidence grid for the hallway environment. Cells representing open space are represented by whitespace. Cells representing occupied space are represented by black circles. Cells representing unknown territory (beyond the hallway walls) are represented by small dots. The five Xs correspond to the robot's starting locations for ARIEL's exploration trials.

The four crosshairs on the map indicate reference points at the corners of the ends of the hallways. Since dead reckoning error accumulates as the robot moves through the world, the points explored last are likely to have the greatest amount of positional error. And since ARIEL always moves to the closest unexplored frontier, one of the ends of the hallways is generally the last place explored. By measuring the difference between the actual position of these hallway corners and the position of these corners in ARIEL's learned maps, the amount of positional error incorporated into the map can be estimated. In these experiments, the maximum error between a reference point and the corresponding feature on the learned grid is used as a bound on the positional error introduced into the map. We refer to this metric as the *reference point error* for an evidence grid.

5.2 Exploration Without Localization

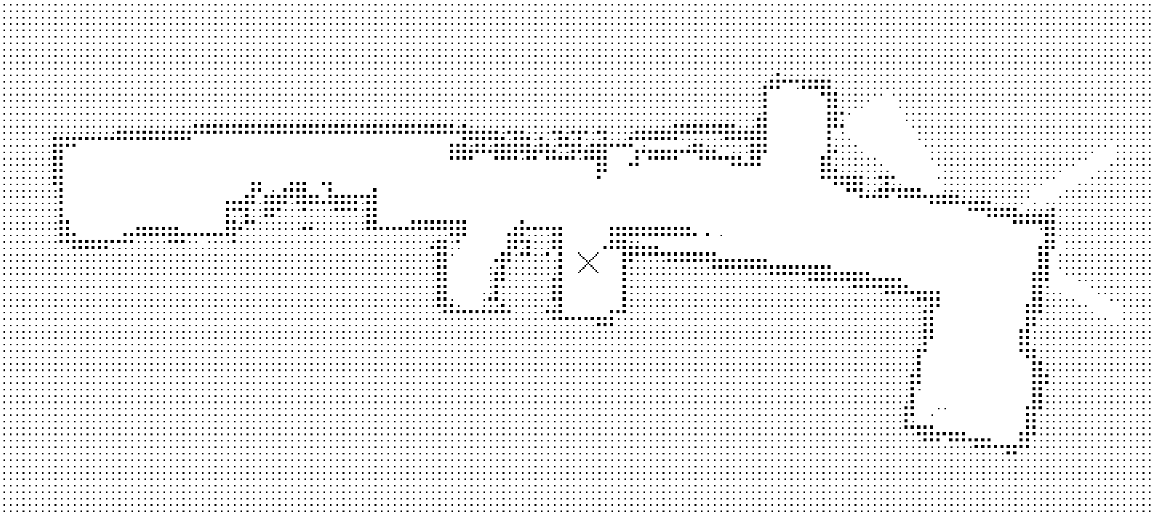

Figure 7: Evidence grid learned without localization

Our first set of trials measured the performance of frontier-based exploration without continuous localization. Five exploration trials were conducted, one from each of the starting locations marked on Figure 6. In three of these trials, frontier-based exploration directed the robot to explore the hallway and build a map, but substantial amounts of position error accumulated during each trial. As a result, sensor information was incorporated into the map at the wrong locations, and the magnitude of this error increased over time.

Figure 7 shows a map learned by frontier-based exploration without localization. The robot started at the position marked with the X. Initially, the robot explored the territory on the left side of the map. Then it navigated back to explore the remaining frontiers on the right side of the map. As the robot explored, position error constantly accumulated. As a result, the right half of the map is considerably more distorted than the left. This grid has a reference point error of 7.0 feet.

In two of the trials, the position error was sufficiently large to prevent further exploration. In both of these cases, the robot started in the middle of the hallway, and explored one side of the hallway first, while remembering the frontier location corresponding to the other side of the hall. When the robot went back to explore the other side, the robot's position error was so large that the relative location of the frontier corresponded to a position behind the (real) hallway walls.

Frontier-based exploration without localization was successful at mapping the entire hallway in 60% of the trials. In the successful trials, the average reference point error for the learned maps was 7.9 feet, and the average amount of time required to explore the hallway was 18.4 minutes.

5.3 Exploration With Localization

Our second set of trials measured ARIEL's performance using frontier-based exploration in combination with continuous localization. We used the same hallway environment, the same starting points for the robot, and the same ground truth evidence grid. Frontier-based exploration again directed the robot to explore the environment, but continuous localization also regularly updated the robot's position estimate as the robot explored. Starting from the same five initial positions shown in Figure 6, ARIEL was able to build a complete map of the environment in all five trials.

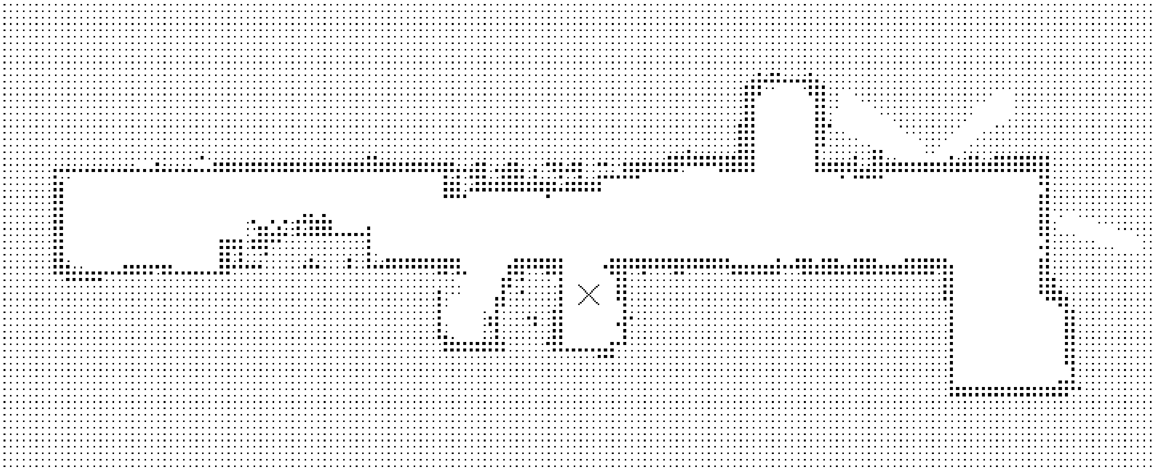

Figure 8: Evidence grid learned using exploration with localization

Figure 8 shows the evidence grid learned using localization starting from the position marked with the X (the same initial position as in Figure 7). In this trial, the robot first explored territory on the right side of the map. Then it navigated back to explore the remaining frontiers on the left side of the map. As the robot explored, continuous localization maintained an accurate position estimate, so the entire environment is mapped correctly. This grid has a reference point error of only 0.4 feet, which is equal to the width of a single grid cell.

ARIEL was successful at mapping the entire hallway in all of the trials. The average reference point error for the learned maps was 2.1 feet, or roughly one quarter of the error in the maps learned without localization. ARIEL’s 100% success rate indicates that this accuracy is sufficient to navigate robustly through this cluttered hallway environment. Reactive obstacle avoidance allows the robot to deal with small errors in the map.

In contrast, exploration without localization had a 40% failure rate, despite having the same capability for reactive obstacle avoidance. This indicates that the accuracy of the maps generated without localization (7.9 foot average reference point error) was not sufficient for robust navigation in this environment.

The average amount of time required to explore the entire hallway was 20.7 minutes. This is slightly longer than the average time (18.4 minutes) required without localization, due to the time required for frontier-based exploration to send its learned evidence grids to continuous localization. However, since the localization process runs on a different processor than the exploration system, the computation required for localization does not slow down the exploration process.

6.0 Related Work

Considerable research has been done in robot map-building, but most of this research has been conducted in simulation [6] or with robots that passively observe the world as they are moved by a human controller [3] [5]. However, a few systems for autonomous exploration have been implemented on real robots.

We previously developed a reactive/topological exploration system for ELDEN (Exploration and Learning in Dynamic ENvironments) [14]. This system had the advantage of being able to adapt its topological map to changes encountered in the environment. However, it also suffered the limitations of a purely reactive exploration strategy, in terms of the size and complexity of the environments that it could explore efficiently.

We also developed a system for place recognition and localization using evidence grids associated with nodes in a topological/metric map [13]. This system had the advantage of being able to localize from a completely unknown initial position. However, the localization process had the disadvantage of being time-consuming. Using a Decstation 5100, this system required approximately 45 seconds to localize within a map containing 20 different places (scaling linearly with the number of places in the map), making this approach unsuitable for continuous localization.

Mataric [8] has developed Toto, a robot that combines reactive exploration, using wall-following and obstacle-avoidance, with a simple topological path planner. The reactive nature of Toto's exploration limits its ability to map environments where wall-following is insufficient to explore the complex structure of the world.

Connell [1] has developed a simple exploration system to demonstrate his SSS architecture. This system was limited to mapping hallways where doors and corridors intersect at 90 degree angles.

Lee [7] has implemented Kuipers Spatial Semantic Hierarchy [6] on a real robot. However, this approach also assumes that all walls are parallel or perpendicular to each other, and this system has only been tested in a simple environment consisting of a three corridors constructed from cardboard barriers.

Thrun and Bücken [11] have developed an exploration system that builds a spatial representation that combines evidence grids with a topological map. This system has been able to explore the network of hallways within a large building. While this approach works well within the hallway domain, it assumes that all walls are either parallel or perpendicular to each other, and that they do not deviate more than 15 degrees from these orientations. An implicit assumption is that walls are observable and not obstructed by obstacles. These assumptions make this approach unsuitable for rooms cluttered with obstacles that may be in arbitrary orientations.

Duckett and Nehmzow [2] have developed a mobile robot system that combines exploration and localization. This system uses a simple reactive wall-following strategy for exploration. For localization, this system uses a self-organizing neural network trained using ART. This network learns to classify regions in space based on input from infrared range sensors. Each of these regions is associated with the corresponding robot position as determined by dead reckoning. By looking at sensor inputs and motor commands over time, this system can reduce the amount of perceptual aliasing that occurs. Since this system relies upon dead reckoning to determine the robot's position during exploration, any drift in dead reckoning during exploration will be incorporated into the map. This robot has only been tested in a small enclosed area (6 meters by 4 meters), so it is unclear whether this approach will scale to larger, more complex, environments.

ARIEL has a number of advantages over previous exploration systems. ARIEL can explore efficiently by moving to the locations that are most likely to add new information to the map. ARIEL can explore environments containing both open and cluttered space, where walls and obstacles are in arbitrary orientations. Finally, ARIEL can maintain an accurate estimate of the robot's position even as it moves into unknown territory.

7.0 Conclusion

We have introduced ARIEL, a mobile robot system that combines frontier-based exploration with continuous localization. ARIEL answers the question of how to learn a new map while simultaneously using that map to self-localize.

Frontier-based exploration is based on the idea of navigating to frontiers, regions on the boundary between space that is known to be open and unknown space. At each frontier, the robot has a clear view into unknown territory. Frontier-based exploration incrementally extends the map as it navigates to each successive frontier. Since this map always includes the territory between the robot and the frontier, continuous localization is able to match the robot's recent perceptions against this map. This match process allows continuous localization to maintain an accurate position estimate as the robot explores the world.

We have conducted experiments in a cluttered hallway from a real-world office environment. These experiments have demonstrated the problems with using dead reckoning for position estimation during exploration. Maps generated using dead reckoning alone tend to contain substantial errors, making them unsuitable for navigation. These experiments have also shown that by integrating frontier-based exploration with continuous localization, ARIEL can explore an unknown environment and build accurate maps that can be used for robust navigation.

8.0 Acknowledgments

This work is supported by the Office of Naval Research.

9.0 References

[1] Jonathan Connell and Sridhar Mahadevan, “Rapid task learning for real robots,” in *Robot Learning*, Jon Connell and Sridhar Mahadevan, Eds., Boston, MA: Kluwer Academic, pp. 105-139, 1993.

[2] Tom Duckett and Ulrich Nehmzow, “Experiments in evidence-based localisation for a mobile robot,” *Proceedings of the AISB Workshop on Spatial Reasoning in Mobile Robots and Animals*, Manchester, UK, 1997.

[3] Sean Engelson, *Passive Map Learning and Visual Place Recognition*, Ph.D. Thesis, Department of Computer Science, Yale University, 1994.

[4] Kevin Graves, Alan Schultz, and William Adams, “Continuous localization in changing environments,” *Proceedings of the 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation*, Monterey, CA, July 1997, pp. 28-33.

[5] David Kortenkamp, *Cognitive Maps for Mobile Robots: A Representation for Mapping and Navigation*, Ph.D. Thesis, Electrical Engineering and Computer Science Department, University of Michigan, 1993.

[6] Benjamin Kuipers and Yung-Tai Byun, “A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations,” *Journal of Robotics and Autonomous Systems*, 8:47-63, 1991.

[7] Wan Yik Lee, *Spatial Semantic Hierarchy for a Physical Robot*, Ph.D. Thesis, Department of Computer Sciences, The University of Texas at Austin, 1996.

[8] Maja Mataric, “Integration of representation into goal-driven behavior-based robots,” *IEEE Transactions on Robotics and Automation*, 8(3):304-312, June 1992.

[9] Hans Moravec and Alberto Elfes, “High resolution maps from wide angle sonar,” *Proceedings of the IEEE International Conference on Robotics and Automation*, St. Louis, MO, 1985, pp. 116-121.

[10] Alan Schultz, William Adams, and John Grefenstette, “Continuous localization using evidence grids,” NCARAI Technical Report AIC-96-007, Naval Research Laboratory, Washington, DC, 1997.

[11] Sebastian Thrun and Arno Bücken, “Integrating grid-based and topological maps for mobile robot navigation,” *Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI-96)*, Portland, OR, August 1996, pp. 944-950.

[12] Brian Yamauchi, “A frontier-based approach for autonomous exploration,” *Proceedings of the 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation*, Monterey, CA, July 1997, pp. 146-151.

[13] Brian Yamauchi and Pat Langley, “Place recognition in dynamic environments,” *Journal of Robotic Systems*, Special Issue on Recent Progress on the Theoretical Developments for Mobile Robots, Vol. 14, No. 2, February 1997, pp. 107-120.

[14] Brian Yamauchi and Randall Beer, “Spatial learning for navigation in dynamic environments,” *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, Special Issue on Learning Autonomous Robots, 26(3):496-505, June 1996.