Multi-Robot Coverage and Exploration: A Survey of Existing Techniques

Brett Walenz
University of Nebraska at Omaha

Abstract

Mobile robots used for coverage and exploration is an important field crucial for tasks such as mine clearing, intrusion detection, sensor deployment, search-and-rescue, and harvesting. In this paper, we cover existing approaches and the evolution of the field. We identify the primary existing techniques, including cellular decomposition, heuristics, insect-based coordination, and multi-robot spanning tree coverage techniques. We also look at a related field for mobile sensor deployment and similarities that the two fields share.

Introduction

Robotic coverage has evolved quickly, from heuristic-based single robot techniques to multiple heterogeneous robots working in a coordinated fashion. Single-robot based techniques have provided inspiration for many of the multiple mobile robot techniques. The Spanning Tree Coverage (STC) algorithm[6] for a single-robot approach breaks down an environment into a grid-based representation and traverses the environment by following a spanning tree of the grid. Boustrophedon decomposition divides the area and allows for non-polygon obstacles to be represented for exploration planning purposes[13][20]. Heuristic approaches have been used to estimate the robots pose within an environment based on the terrain and odometry readings[19].

Multi-robot approaches started with heuristic and rule-based approaches. [21] used pairs of robots to correct for odometry errors, where one robot would monitor the exploring robot and ensure correct readings. Batalin implemented two simple local heuristics for dispersing a team of mobile robots for sensor coverage [1][2]. Biological inspired approaches look for solutions amongst communities of insects that are common to multi-robot issues. Wagner et al. [24] took inspiration from biology and used an ant-based approach where a team of mobile robots would leave scent trails for recently explored areas.

Spanning tree coverage approaches build upon the single robot technique and create robust, distributed multi-robot spanning tree coverage algorithms, producing the Multi-robot Spanning Tree Coverage (MSTC) and Multi-Robot Forest Coverage (MFC) algorithms. Probabilistic approaches exploit posterior information coupled with environmental information to reason about the robots situation. Burgard[3][4] used a probabilistic mechanism that took into account the cost of traveling to the next exploration area with the utility gained by using the robots sensors. Thrun used maximum likelihood estimation to calculate the robots local pose and aimed to reduce the error for odometry readings.

This survey paper will cover the common approaches in multi-robot coverage and exploration and cover recent advances in the primary approaches used. Multi-robot coverage and exploration strategies can be broken down into different areas, although recent approaches overlap and combine existing techniques. For the different techniques, an in-depth overview of the primary principles of the technique will be provided, followed by an overview of recent extensions to the technique.

Heuristic and Biological Approaches

Initial multi-robot exploration techniques were simplistic, and relied primarily on limited communication approaches. Rekleitis [19][21] used teams of robots for collaborative exploration with the intent of reducing the errors due to odometer readings. The interaction between pairs of robots can be broken down by their local and global interactions. In a global sense, the robots work in tandem by first breaking down the area to be explored into trapezoids. The order of exploration of the trapezoids is determined by a depth first traversal of the graph connections of the trapezoids. At the local stage, a trapezoid is broken down into stripes taking into account the robot's sensing range. One robot traverses the strip while the other robot is stationary. The stationary robot is responsible for tracking and monitoring the moving robots trajectory, maintaining an uninterrupted line of sight. If the line of sight is interrupted, the mobile robot backtracks and finds an alternate route around the obstacle. This method permits large-scale exploration by reducing the error introduced by the frictional properties of certain terrain, which could cause odometry errors that accumulate over a large distance.

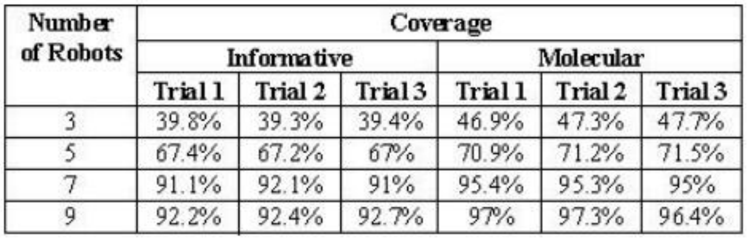
While the above description describes the interactions between two robots, Rekleitis also describes the case of *n* robots. Two strategies are outlined and differ on the basis of separate coordinated teams. In the first strategy, the robots function as one united team with the goal of reducing uncertainty. In the second strategy, robots divide into two teams and switch roles based on position within the current trapezoid space. One team moves through the region and functions as the exploring robots, while the other team functions as landmarks and provides relative readings and odometry corrections. In the first case, only one robot is mobile and the rest of the robots are considered the observers.

Batalin proposed two simple heuristics for distributing a team of mobile robots for sensor coverage[1]. Both algorithms operate at a local scale and aim at dispersing the local network with the intent of influencing the overall global structure of the robots. The first approach, called *Informative*, has robots spread out in a coordinated fashion. Each robot assumes an identity within the group and relative positions are calculated. The robots then disperse to maximize the groups coverage. The second approach, *Molecular*, is more simplistic. Robots act individually and disperse in a repulsive manner by calculating the vector sum in the opposite direction of all the other robots. Batalin compares these algorithms with a Basic approach, which only seeks to maximize each individual robots sensor coverage.

The experiments focused on varying two parameters: team size and starting positions. Robots in this setting are considered homogeneous with four behaviors: Obstacle Avoidance, Walk, Observe, and Dance. The Observe behavior selects the most 'promising' vector for exploration. The Dance behavior is an algorithm specific implementation which determines the means by which robots locally disperse. In the case of the *Informative* approach, robots coordinate with each other to maximize their local network and can send each other Dance messages.

The reception of this message causes the robot to enter an observer state where the robot observes the dancing robot and determines a relative position and calculates the ideal positioning with respect to the dancing robot. In the *Molecular* approach, direct communication is not used, instead only vision is used. The dancer selects a position which is in the opposite direction of the average angle of all its neighbors - essentially creating a repulsive potential field from nearby robots.

The experimental results of this paper show that even minimal interactions at a local scale are extremely useful for coverage of an unknown area[2]. Both *Informative* and *Molecular* approaches perform substantially better then the Basic approach, and the coverage area quickly maximizes with the addition of a few robots. An important result is that even though the *Molecular* approach is more simplistic, it outperformed the *Informative* approach in the experiments. Batalin theorizes that this is due to the amount of overhead added by the Informative approaches determination of relative positions amongst robots.



Number of Robots	Coverage					
	Informative			Molecular		
	Trial 1	Trial 2	Trial 3	Trial 1	Trial 2	Trial 3
3	39.8%	39.3%	39.4%	46.9%	47.3%	47.7%
5	67.4%	67.2%	67%	70.9%	71.2%	71.5%
7	91.1%	92.1%	91%	95.4%	95.3%	95%
9	92.2%	92.4%	92.7%	97%	97.3%	96.4%

Figure 1 - Experimental Results of Informative and Molecular Approaches

Wagner et al. [24] advanced the field of robotic coverage in 1996 with their introduction of using an ant-based strategy. Taking inspiration from the way ants alert each other to the presence of food through the use of pheromones, or chemical signals that trigger responses in other ants, Wagner et al. equipped simulated robots with the same device for coverage and exploration tasks. Upon entering a cell, a robot would deposit a pheromone which could then be detected by other robots. These pheromones had a decay rate associated with them, allowing continuous coverage of an area via implicit coordination.

Their initial approach used a cellular decomposition of the explored space with three different ant-based strategies. The first rule, ANT-WALK-1, is a local rule that requires no individual memory on the robots. Shared memory is composed of the smell traces that are laid on the edges of the graph G (the connections between cells in the decomposition of the space). ANT-WALK-2 uses a Depth-First-Search methodology but modified to work in a multi-robot setting. In the basic case, each robot has a limited memory allowing robots to keep track of vertices visited and how they got to each vertex to satisfy the DFS requirements for backtracking. VERTEX-WALK-1 leaves pheromone traces at the vertices, rather than the edges.

Wagner et al. then used a simulated environment to test the different strategies. For each strategy, the ‘noise’ level is varied, where noise is defined as a frequency function on the pheromone at a given position. Their results indicate that the ANT-WALK strategies are far better in noisy situations and that VERTEX-ANT-WALK performs better when no noise is present. This ant-based approach has inspired many similar approaches and was an upgrade to existing approaches in the adaptivity and robustness of the system.

Probabilistic Approaches

Simmons et al. [22] used a semi-distributed model where a centralized module integrated local data from a team of mobile robots. The team used a probabilistic technique to build a global map in a coordinated fashion. Due to the problem of absolute positioning techniques in indoor environments, robots must estimate their local pose in an environment, leading to odometry errors. Simmons used probability calculations to estimate the local pose of each robot, and built a global map by joining each individual robots local map. Simmons’s approach centers around using maximum likelihood estimation, a statistical method use for fitting model to data. Maps are created in an online fashion by using an estimate of the robots previous pose combined with the most recent sensor measurement.

Simmons work was further defined by Thrun [23], but still faces two restrictions: the robots must begin nearby and within sensor range so their scans contain overlap, and the software must be told of their initial starting relative positions. Each robot stores a local estimate of its pose, but all robots use the same map. The estimation of the most likely pose is based on the single map that combines all the data from the team of robots. Effectively, each robot performs the same basic estimation algorithm to estimate its pose, but integrated map results are broadcast to all robots.

As a more in-depth look, each robot maintains maximum likelihood estimates for its position, the map (as a result of the scan by the sensor), and a density function for the potential true locations. This density function is a characterization of the possible locations the robot could have come from, indicating that uncertainty is an inherent mechanism in the system. Using these probability densities, a special function is used to find the maximum likelihood for each of the readings, and a resultant map is built. This map is converted to an occupancy grid map that is commonly used in robotic coverage.

As mentioned above, integrated map results are broadcast to all robots, however individual maps are merged together by a central mechanism. The central mechanism operates on the same principles as each individual robot by merging the probabilities of individual readings and selecting the maximum likelihood.

While Thrun expanded the basic probabilistic model, the original concept proposed in Simmons used an auction mechanism for assigning exploration tasks, again using the concept of frontiers. Similar to other approaches, bids are calculating using cost and information gain, where cost is defined as the distance required to reach the frontier zone, and information gain is the amount of exploration the robot is capable of generating upon reaching the zone. Bidding is controlled by a central auctioneer, which assigns tasks by assigning locations with the highest net utility first. In both Simmons and Thrun's work, a central mechanism is necessary for map integration and the auction mechanism.

Thrun's work experimentally validated this approach using real robots. In particular, Thrun's work is notable because it used heterogeneous robots with a high rate of error. Two robots are of the pioneer variety, while a third robot is an urban robot with a high degree of uncertainty in its sensors. Even with the third robot functioning at a lower scale, the system is able to adapt and create a well detailed map.

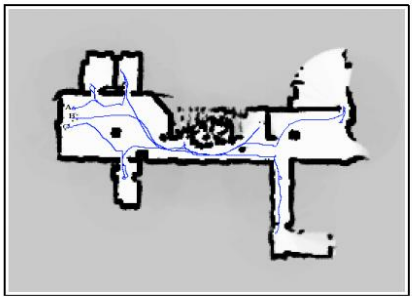


Figure 2 - Map generated from maximum likelihood estimation

Yamauchi's *Frontier Based Exploration Using Mobile Robots* [25] is a foundational paper that is used by many successful approaches for multi-robot coverage and exploration. In this approach, robots act independently and make probabilistic judgements regarding frontiers – areas of unexplored space in an environment. The environment is decomposed into cells with each cell represented by a probability value, which is represented by one of three classes: open, unknown, or occupied. A robot will navigate to the nearest unvisited frontier, and upon arrival will perform a sensor sweep.

Upon completion of the sweep the robot will broadcast the results to the other robots, who will then incorporate the results by adding probabilities together, creating a local map for each robot. This approach is cooperative and distributed, with no central mechanism. Using this algorithm, independent observations can be merged together. A robot can then recognize frontiers that have been detected by others robots and can explore those areas. While information is shared between robots, it is not a requirement for successful exploration of the environment, making the system robust and able to handle failures within the robot team.

Yamauchi's approach is an important approach due to the fully distributed nature of the approach. In contrast to the work by Simmons et al., which requires a central mechanism to handle map integration and auctions, Yamauchi's probabilistic mechanism for map integration is efficient and runs on-board. However, Yamauchi's research is also important for another reason. During the implementation of the algorithm in a live robot environment, Yamauchi initially used dead reckoning. Unfortunately, this lead to large odometry errors and Yamauchi also used a local positioning estimation function. Later on in this survey, we will see how Yamauchi's frontier-based methodology is integrated with Simmons maximum likelihood pose estimation techniques for use in a large scale distributed exploration environment [18].

Similar to the work proposed by Simmons, Burgard[3] tackles the multi-robot exploration problem by considering the exploration function as a probabilistic mechanism dependent on the cost of travel and the utility gained by the robots sensors. Burgard's approach uses occupancy grid maps and frontier cells to calculate the cost of each robot traveling to each frontier cell. The utility of the robot exploring the frontier cell is based on the distance the robot can cover with its sensors. It is assumed that robots know each others relative positions.

The algorithm determines the optimal target points for each robot that increase the coverage by the maximum amount at that time period. Once a frontier cell is assigned to a specific robot, the utility of that target point is reduced by a factor of the exploring robots utility. In other words, the utility of a target point is related to the expected utility that a robot provides to the overall coverage problem by using its sensors at the specified target point.

Burgard uses an occupancy grid map and merges corresponsing occupancy grid maps from separate robots by using probabilities. The idea here is that in each cell the probability that the cell is occupied by an obstacle is denoted by $P(occ_{x,y})$. If each robot has an occupancy map, Burgard defines the merged, global occupancy map by the following formulas:

$$P(occ_{x,y}) = \frac{odds_{x,y}}{1 + odds_{x,y}} \qquad (1)$$

where

$$odds_{x,y} = \prod_{i=1}^{n} odds^{i}_{x,y} \qquad (2)$$

and

$$odds^{i}_{x,y} = \frac{P(occ^{i}_{x,y})}{1 - P(occ^{i}_{x,y})} \qquad (3)$$

Figure 3 - Burgard's Occupancy Grid Merging

This is most easily explained by starting at equation (3). The odds of *x,y* being occupied in robot *i*'s map is given by the ratio of the probability that *x,y* is occupied according to *i* over the probability that it is not occupied. Therefore, if *i* doesn't think *x,y* is occupied, the odds will be close to 0. Then, equation (2) calculates the odds that the entire team thinks *x,y* is occupied. Thus, equation (1) gives the odds at a global scale that *x,y* is occupied by an obstacle.

Burgard's initial approach uses a valuation technique defined as $P = (R - C)$, where R is defined as the expected information gain from the system by using robot *i*'s sensors at the target point, and C is defined as the expected cost of traveling to the target point. The allocation strategy then involves selecting the max P for each target point, and an update mechanism to reduce the expected utility of other robots to avoid overlap. In [4], Burgard expands on the initial problem by using a bidding strategy where different robots bid for the opportunity to explore a target point. A central arbiter then chooses the optimal bid set.

Experimental results indicate that this methodology reduces the coverage time for the coordinated case vs the uncoordinated case. In a simulated environment, a team of 8 robots completed the exploration task working together in 10 minutes, while in the uncoordinated case the completion time was 16 minutes.

Relatively close to Burgard's approach in *Collaborative Multi-Robot Exploration*, Zlot [27] uses the concept of frontier cells and utility in a market environment to produce complex coordinated behavior between multi-robots for exploration. In this approach, exploration is performed by visiting goal points (referred to as target points in Burgard's work) in unexplored regions. A robot contains a task list called a tour which contains several goal points. Tours are refined through robot negotiation in the market, and allows the robots to improve their exploration in an efficient manner. Each robot has a market agent called the operator executive, or OpExec, which is responsible for processing bids, handling revenue, and so forth. The revenue function is similar to Burgard's, where $P = R - C$.

Tasks are the primary commodity handled in the market. By definition, a task is a singular goal point to visit, thus a key component of the system is the methodology by which goal points are selected. Four different strategies are analyzed: random, greedy, quadtree division, and no communication. In the random strategy, goal points are selected by a random process, but any goal points are discarded if the area surrounding the goal point has been visited. The greedy process chooses the goal point in the closest unexplored region to the robot. The quadtree strategy transforms the space into a quadtree representation, and the largest unknown leaf regions are selected as candidates for exploration, with goal points located at the center of the leaf. The no communication scenario emulates robots with no coordination available. In this case, robots used the random goal generation strategy without any information sharing.

Robots negotiate using single-item first-price sealed-bid auctions, and may auction any of their tasks in their current tour. The robots internal valuation of the task is the profit expected if the task was added to the robots existing tour, and the auctioneer then announces a reservation price P_r, where P_r is the robots valuation plus a small fixed amount. Bidding strategies for robots are defined by the formula:

$$B_i = P_r + \alpha * (v_i - P_r)$$

Figure 4 - Bid Mechanism

Where *a* is between 0 and 1, with higher coefficients resulting in an increased incentive to sell the item, while at the same time the buyer gets a large fraction of their valuation if they win. This enables the robot to effectively sell an item to the robot that can perform the task most efficiently.

The system is responsive to robot loss due to the nature of the goal generation strategy. If a robot fails while completing a tour, existing robots will eventually generate goal points in those areas, thereby negating the loss of that robot. In addition, map sharing is done at explicit intervals (and can be manually performed by a human operator), where a robot periodically sends out a small explored section of its own map on a peer-to-peer basis with robots in communication range. The maps are composed in a simplistic manner whereas [3] uses a probabilistic mechanism to combine occupancy maps. Experimental results indicate that the market based economy functions much better than a non-coordinated approach. Both random and quadtree goal point selection strategies have an area covered to distance traveled ratio of greater than 1, while the no communication strategy ratio is significantly less than 1, meaning that robots are covering less area than they are traveling.

Spanning Tree Coverage Approaches

In single robot uses, STC is guaranteed to provide complete coverage[10][11]. Hazon and Kaminka[14] adapted the STC to the multi-robot situation and created the Multi-robot Spanning Tree Coverage (MSTC) algorithm. The initial approach is an off-line approach: it's assumed that the robots have a map of the area *a priori*. MSTC decomposes the area into cells of size 4D, where D is the sensor range of the robots in the simulation. Each cell is further broken down into quadrants of size D. MSTC works by first constructing a spanning tree *S* for the graph *G* which covers the environment. *S* is then broken down into sections, S_0 ... S_n, where *n* is the number of robots in the system. Each robot is then assigned a section and covers that specific section in a counterclockwise fashion. If a robot does not finish or does not respond after a specified period, that robot is assumed to have failed and the robot in the section behind will pick up that portion of the work.

In *Towards Robust On-line Multi-Robot Coverage* [9], the authors discuss an on-line spanning tree coverage (STC) approach called ORMSTC – On-Line Robust Multi-robot Spanning Tree Coverage. The robots work by independently executing STC coverage algorithms, but check with nearby robots upon entering a neighboring cell. If the cell is already marked by an existing robot and that robot is alive, the robot will proceed in a different direction. The goal of the algorithm is to ensure coverage completes, even with multiple robot failures.

Similar to MSTC, ORMSTC decomposes the area into cells of size 4D, where D is the sensor range of the robots in the simulation. Each cell is further broken down into quadrants of size D. Each robot builds a local spanning tree through the exploration process, and tracks the state of other robots (that it has come across) within the area.

When a robot first meets a cell that overlaps with another robots coverage tree, the robot denotes this and saves its connections with other robots. In this fashion, robots do not have to keep track of the entire region and do not have large information sharing requirements. Furthermore, when a robot enters a new region that has been marked again by a robot it is already tracking, the connection tree is updated.

While a robot is checking connections, if the robot determines that another robot *j* is not alive or not responsive, it broadcasts to other robots this fact, and the cells that robot *j* was marked as connected are now marked as empty. This allows other robots to map the area and take over for the fallen robot. The coverage process is determined to be finished when all robots have completed their explorations.

Experimental simulations of ORMSTC have shown its effectiveness. Using Player/Stage[12], the researchers focused on two simulated environments: a cave, which was relatively open but had objects of various shapes, and a room which had many rectangular obstacles. The robotic team size was varied between 2, 4, 6, 8, and 10. In this case, they were interested in measuring the time for complete coverage rather than a percentage of coverage over time. With 2 robots, the system requires nearly 13000 seconds for complete coverage, whereas with 10 robots the system requires only 4000 seconds, regardless of room type.

Zheng et al. [26] builds upon the work in [14] by a variation on the MSTC algorithm called Multi-Robot Forest Coverage (MFC). As in the MSTC approach, the terrain is represented by a cellular decomposition into large cells, with each cell consisting of 4 smaller subcells. MFC is a polynomial-time multi-robot coverage heuristic that uses algorithms for finding a tree cover from a given set of roots. A tree cover is defined as follows. Assume a graph $G = (V, E)$ and $R \in V$ be a set of roots. A tree cover of G is a forest of size R trees that cover V. The tree cover algorithm is a polynomial time algorithm that approximates the optimal tree covering.

Using the tree cover algorithm, the current positions of the robots are treated as the set R, and the discovered tree covering is the spanning tree for each robot. Thus, the tree covering algorithm allows robots to distribute the exploration workload in a near-optimal fashion while permitting overlap in the tree exploration.

Zheng et al. show that the MFC algorithm performs much better than MSTC for large numbers of robots. In fact, using the MSTC algorithm the research shows that with large numbers of robots the ratio of cover time to ideal cover time increases very quickly, while for MFC the ratio increases very slowly, indicating that MFC scales quite well. These results are seen for both “cover”, where the robots merely have to cover the area, and “cover and return” where the robots must cover an area and end up back at their starting position. Zheng et al. theorize that MSTC performs poorly due to the inability of MSTC to fully balance the travel costs of the robots, whereas MFC fully permits overlapping paths for efficiency purposes.

Elmaliach[8] uses the MSTC algorithm for a team of mobile robots for area patrol. They define the problem as repeatedly visiting a set of target locations to assess the environmental status, while at the same ensuring the system is robust, so even if robots fail coverage of the area is still guaranteed (although the frequency of coverage will drop). Minor extensions to the MSTC algorithm are provided here. First, the authors improve the algorithm by account for directionally-aware cost metrics by asserting a robotic velocity differential based on the terrain and direction they travel. They then provide an algorithm to generate the minimal cyclic path given this terrain situation, and position the robots optimally along this minimal path to provide ideal coverage at the desired frequency.

Elmaliach identifies the frequency variable as a primary key in determining the behavior of the algorithm, and defines three different metrics for area coverage:

1. Uniform frequency – all targets in the area should be covered with the same frequency *f*
2. Average frequency – all targets in the area should be covered with an average frequency *f*
3. Under-bounded frequency – all targets should be covered at least once every *1/f* cycles

Their research generates an algorithm that can guarantee uniform, under-bounded, and average frequency. By placing robots with an equal spacing apart, the researchers can guarantee that all three metrics are satisfied, and prove so.

Hybrid Approaches

[18] makes a substantial contribution to the field of multi-robotic coverage and exploration by assembling a team of 80 heterogeneous robots and performing repeated experiments. The system was built to address the challenge of a multi-robot team which could explore a large environment and set up a stable sensor network capable of identifying and tracking intruders. Due to cost, there team consists of two types of robots. A small number of leader robots, equipped with a laser range finder and on-board CPU, these robots are responsible for leading the smaller group of robots to their designated positions and maintaining the map of the environment. The other group is simple robots which have a microphone and basic camera, but are unable to fully navigate themselves through the environment. These simple robots are placed in the environment by the leaders but have enough knowledge to identify and track intruders given their limited sensors.

Figure 5 - The multi-robot team used in [18]. Two types of robots can be seen: leader robots and sensor robots.

The above task can be broken down into two sections: mapping and exploration, and intruder detection in a distributed sensor network. The former will not be covered in this paper. For exploration, the system uses a decentralized frontier-based approach with local occupancy grids;

there is little communication amongst robots. Following the frontier-based approach, the system works as follows:

1. Construct a local occupancy grid. This is done by using the laser range data and local pose estimates, which is implemented using the approach described in [23]
2. Identify the frontiers in the local occupancy grid.
3. If a frontier is no longer available due to obstruction, select another one.

Unfortunately, this approach suffers a similar fate as its predecessors. The use of a local occupancy grid introduces a situation where a robot may explore the same area multiple times (due to forgetting old explored areas), and the lack of robot communication leads to robots overlapping coverage areas. Varying the team size has little effect: while the coverage rises very quickly at start, their experiments show that regardless of team size, there is a slow convergence to full coverage. The advantage of this approach, however, is that the resultant map is usually quite good. Experimental results indicate that the robotic team achieved accuracy that was comparable to a human survey team (which was done with pen and paper).

In *Distributed Coverage of Rectilinear Environments*[5], Butler uses a cellular decomposition approach coupled with a "seed-sowing" methodology for navigating through the cells. Each cell is traversed by a seed-sowing approach, where the robot navigates in stripes ensuring complete coverage of the cell. Robots maintain their own internal map **C**, and event handlers monitor incoming transmissions. Updates to **C** from other robots are handled seamlessly, separating the distributed network from the exploration function. Butler proves that this approach produces complete coverage in both single robot and multiple robot cases.

Multi-robot coordination works as follows. Suppose a robot i identifies a new cell, C_{new}. An overseer handler is responsible for integrating this incoming information into robot j's existing **C**. This is done in three stages. In the first stage, new cells are added to **C** that are the dimensions of C_{new}. Then, for each cell C_{new} in the incoming update, the added cells in **C** are altered so they do not overlap. Lastly, the new cells are merged with the existing map, creating walls, neighbors, or placeholders for exploration. The necessity of this three stage process is driven by the autonomy of the robots with the potential to generate a separate cellular decomposition of the environment as they explore different regions.

Butler's experimental results are especially interesting because of the experimental constraints – positioning and collisions. Butler tested various conditions, the first of which was starting position. In one case, robots were positioned parallel to each other, while in the other case they started perpendicular. Butler's research is also notable for the fact that he tested robots that were able to avoid collisions with each other but in some experiments were not. He also varied the number of teams, using 2, 3, 5, and 10 robot teams. In all cases, results were similar, although there are differences between the positioning and collision cases. Comparisons were made against a baseline of a single-robot in terms of efficiency (with the multi robot team completion was asserted when they had reached the same coverage percentage as the single-robot ended up with).

Essentially, Butler showed that increasing the number of teams showed substantial increases in efficiency. First, two robot teams generally had to travel about 65% the distance that a single robot did, and the coverage time was generally reduced by 20-25%. As mentioned earlier, an interesting side-effect is that orientation matters. In the two robot case, it appears that perpendicular orientations are more efficient, and Butler speculates this is because orientation matters when generating maps, so parallel orientations will lead to one robot doing more work than the other robot. In the case of with collisions vs. without collisions, the results were expected. Without collisions, the teams explored the environment more efficiently than those with collisions. However, an important result is that the performance is not largely different – teams of multiple robots that could collide with each other still functioned adequately compared to the single-robot case.

Related Applications

A closely related topic to mobile robotic coverage is the deployment of a mobile sensor network. In this scenario, a team of robots is assigned the task of establishing a sensor network that provides complete coverage of an area. For instance, mobile robots may be armed with chemical sensors during a chemical spill to alert damage control teams of safe or unsafe conditions within an environment. While the two areas are somewhat different, the techniques often overlap and it's worthwhile to illustrate related use cases[7]. We've already looked at a few related applications that closely overlap multi-robotic coverage and exploration. In particular, the approaches by Batalin et al.[1][2] have a high degree of overlap.

Howard[15] uses a traditional approach of potential fields in robotics and enhances it for the purposes of deployment of mobile sensor networks. Robots are subjected to virtual repulsive forces from other robots and an internal viscuous force that ensures equilibrium is eventually reached. Obstacles are likewise treated as repulsive forces, allowing robots to deploy around obstacles while maintaining an equilibrium state with other robots. The viscuous force is not an overriding force within the system, rather it is a stabilizing force. Changes in the environment will override this stabilizing force, causing the network to adjust but ensuring that it eventually stabilizes. The approach is an entirely local approach, and does not require global positioning or localization techniques.

Simulations were run using the Player/Stage environment, and measured two metrics: coverage (amount of the environment monitored by the sensor network) and time (amount of time to reach the maximum coverage). The environment is a large-scale office environment with multiple rooms, and the system performs quite well. Coverage increases quickly – using 100 robots the robots are deployed at the start to 50 m^2, and after 300 seconds coverage increases to 500 m^2. Furthermore, even without explicit coordination, robots are evenly spaced.

Conclusion

Multi-robot coverage has uses in mine sweeping, intrusion detection, and many other industrial applications. In this paper, we have looked at recent approaches that have drawn technology closer to realizing the goals of these uses. Many methods are based on cellular decomposition and information sharing. Probabilistic approaches are used to overcome odometry errors and local pose readings for distributed map building. Recent hybrid approaches combine multi-robot spanning tree coverage algorithms with error correction algorithms, and recent experiments have used upwards of 80 robots for intrusion detection. We have also looked at a related field for mobile network deployment, where the problem is establishing a sensor network capable of monitoring an environment. In both cases of coverage and deployment, the results are encouraging, with recent techniques evolving from and combining earlier techniques.

References

[1]Batalin, M. A. and Sukhatme, G. S. (2002). Spreading out: A local approach to multi-robot coverage. In *in Proc. of 6th International Symposium on Distributed Autonomous Robotic Systems*, volume 5, pages 373–382.

[2]Batalin, M. A. and Sukhatme, G. S. (2003). Coverage, exploration and deployment by a mobile robot and communication network. In *Telecommunication Systems Journal, Special Issue on Wireless Sensor Networks*, volume 26, pages 376–391.

[3]Burgard, W., Moors, M., Fox, D., Simmons, R., and Thrun, S. (2000). Collaborative multi-robot exploration.

[4]Burgard, W., Moors, M., Stachniss, C., and Schneider, F. (2005). Coordinated multi-robot exploration. *IEEE Transactions on Robotics*, 21:376–386.

[5]Butler, Z. J. and Zack (2000). Distributed coverage of rectilinear environments. In *in Proc. of the Workshop on the Algorithmic Foundations of Robotics*.

[6]Choset, H. (2001). Coverage for robotics – a survey of recent results. *Annals of Mathematics and Artificial Intelligence*, 31(1-4):113–126.

[7]Cortes, J., Martinez, S., Karatas, T., and Bullo, F. (2004). Coverage control for mobile sensing networks. *Robotics and Automation, IEEE Transactions on*, 20(2):243–255.

[8]Elmaliach, Y., Agmon, N., and Kaminka, G. A. (2007). Multi-robot area patrol under frequency constraints. *ICRA*.

[9]Fabrizio, N. H. Towards robust on-line multi-robot coverage.

[10]Gabriely, Y. and Rimon, E. (2001). Spanning-tree based coverage of continuous areas by a mobile robot. *Annals of Mathematics and Artificial Intelligence*, 31(1-4):77–98.

[11]Gabriely, Y. and Rimon, E. (2003). Competitive on-line coverage of grid environments by a mobile robot. *Comput. Geom. Theory Appl.*, 24(3):197–224.

[12]Gerkey, B. P., Vaughan, R. T., and Howard, A. (2003). The player/stage project: Tools for multi-robot and distributed sensor systems. In *In Proceedings of the 11th International Conference on Advanced Robotics*, pages 317–323.

[13]Grabowski, R., Navarro-serment, L. E., Paredis, C. J. J., and Khosla, P. K. (2000). Heterogeneous teams of modular robots for mapping and exploration. *Autonomous Robots,* 8:293–308.

[14]Hazon, N. and Kaminka, G. A. Redundancy, efficiency and robustness in multi-robot coverage.

[15]Howard, A., Mataric, M. J., and Sukhatme, G. S. (2002). Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem.

[16]Jiao, L. and Tang, Z. (2008). A visibility-based algorithm for multi-robot boundary coverage. *International Journal of Advanced Robotic Systems*.

[17]Kurabayashi, D., Ota, J., and Yoshida, E. (1995). An algorithm of dividing a work area to multiple mobile robots. In *IROS '95: Proceedings of the International Conference on Intelligent Robots and Systems-Volume 2,* pages 2286+, Washington, DC, USA. IEEE Computer Society.

[18]Parker, L. E. and Howard, A. (2006). Experiments with a large heterogeneous mobile robot team: Exploration, mapping, deployment and detection. *International Journal of Robotics Research*, 25:431–447.

[19]Rekleitis, I., Dudek, G., and Milios, E. (2001). Multi-robot collaboration for robust exploration. *Annals of Mathematics and Artificial Intelligence,* 31(1-4):7–40.

[20]Rekleitis, I., New, A. P., Rankin, E. S., and Choset, H. (2008). Efficient boustrophedon multi-robot coverage: an algorithmic approach. *Annals of Mathematics and Artificial Intelligence,* 52(2-4):109–142.

[21]Rekleitis, I. M., Dudek, G., and Milios, E. E. (1997). Multi-robot exploration of an unknown environment, efficiently reducing the odometry error. In *IJCAI'97: Proceedings of the Fifteenth international joint conference on Artifical intelligence,* pages 1340–1345, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

[22]Simmons, R., Apfelbaum, D., Burgard, W., and Moors, D. F. M. (2000). Coordination for multi-robot exploration and mapping. In *In Proceedings of the AAAI National Conference on Artificial Intelligence*, pages 852–858.

[23]Thrun, S. (2001). An online mapping algorithm for teams of mobile robots. *International Journal of Robotics Research*, 20.

[24]Wagner, I. A., Lindenbaum, M., and Bruckstein, A. M. (1999). Distributed covering by ant-robots using evaporating traces. *IEEE Transactions on Robotics and Automation*, 15:918–933.

[25]Yamauchi, B. (1998). Frontier-based exploration using multiple robots. In *AGENTS ’98: Proceedings of the second international conference on Autonomous agents*, pages 47–53, New York, NY, USA. ACM.

[26]Zheng, X., Jain, S., Koenig, S., and Kempe, D. (2005). Multi-robot forest coverage. In *In Proc. IROS*, pages 3852–3857.

[27]Zlot, R., Stentz, A. T., Dias, M. B., and Thayer, S. (2002). Multi-robot exploration controlled by a market economy.