COORDINATING MULTIPLE ROBOTS DURING EXPLORATION UNDER COMMUNICATION WITH LIMITED BANDWIDTH

Daniel Meier *Cyrill Stachniss* *Wolfram Burgard*

University of Freiburg, Department of Computer Science, D-79110 Freiburg, Germany

ABSTRACT

In this paper, we consider the problem of exploring an unknown environment with a team of mobile robots using a communication link with limited bandwidth. The key problem to be solved in this context is to decide which information should be transmitted over the network to enable the robots to choose appropriate target points. In this paper, each robot approximates its map by a set of polygons. We present an efficient way to incrementally improve the representation. We furthermore adapt an existing coordination strategy so that it is able to deal with our approximated maps in a distributed fashion. Our technique has been implemented and tested. The results demonstrate that our distributed technique can efficiently spread the robots in the environment even if the communication link provides only a low bandwidth. As a result, the robots are able to quickly accomplish their exploration mission despite the constraints introduced by the limited bandwidth.

1. INTRODUCTION

Exploring an environment belongs to the fundamental problems in mobile robotics. There are several applications like planetary exploration [1], rescue [16, 18], mowing [11], or cleaning [13, 17] in which the complete coverage of a terrain belongs to the integral parts of a robotic mission.

To efficiently accomplish an exploration task with multiple robots, a coordination strategy is needed to assign target locations to the different robots [4]. To implement such a technique using a team of real robots, a fast network connection is required in order to send the environmental information to each robot. In real applications, all sensor measurements need to be sent to all other robots or the whole team has to exchange the map. Whereas this is feasible for small groups of robots, it introduces a serious communication overhead for big teams. Therefore, taking the available bandwidth into account when exchanging information is an important requirement for larger robot teams.

In this work, we present a distributed approach to multi-robot exploration for situations in which the network connections have unlimited range and are reliable but only provide a limited bandwidth. Our algorithm computes a polygonal approximation for each map learned by a robot and transmits only changes and refinements of this map to the other robots over the communication link. Based on their own maps and the approximated descriptions of the areas covered by the other robots, each team mate chooses a target location it plans to attain. It then broadcasts this location to the other robots, which consider this plan when calculating their own target location.

2. RELATED WORK

The various aspects of the problem of exploring unknown environments with teams of mobile robots have been studied intensively in the past. For example, Yamauchi *et al.* [21] present a technique to learn maps with a team of mobile robots. In their approach, the robots exchange information about the map that is continuously updated whenever new sensor input arrives. They furthermore introduced the idea of a frontier, which separates the environment into known and unknown areas. Burgard *et al.* [4] presented a technique to coordinate teams of mobile robots which extends a work published in 2000 [3]. Their approach trades off the cost of moving to frontiers with the expected amount of information that can be obtained when a robot arrives at that frontier. Ko *et al.* [14] apply a similar coordination technique that uses the Hungarian Method [15] to compute the assignments of frontier cells to robots. Howard *et al.* [10] presented an incremental deployment approach that aims to coordinate the robots in a similar way. Zlot *et al.* [22] as well as Gerkey and Matarić [7] have proposed an architecture for mobile robot teams in which the exploration is guided by a market economy. Their approach trades tasks using single-item first-price sealed-bid auctions between the robots. In these approaches, it is typically assumed that the network connections have a sufficiently high bandwidth. In contrast to that, the algorithm proposed in this paper is designed to deal with low bandwidth communication links.

In [2], Balch and Arkin analyze the effects of different kinds of communication on the performance of teams of mobile robots that perform tasks like searching for objects or covering a terrain. The "graze task" carried out by the team of robots corresponds to an exploration behavior. One

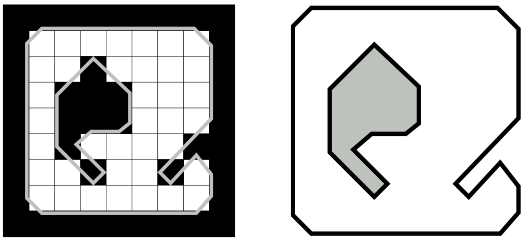

Figure 1: This figure shows the polygon map (right) that is derived by our algorithm from the grid map (left).

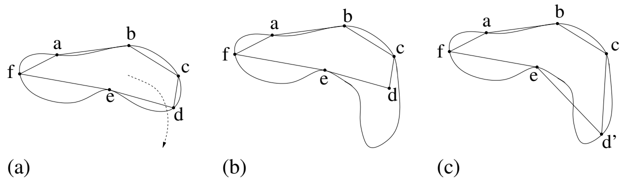

Figure 2: Case 1: A vertex, which was located on $\mathcal{B}_{t-1}$ is no longer located on $\mathcal{B}_t$ due to newly observed areas.

of the results is that the communication of goal locations does not help if the robots can detect the "graze swathes" of other robots. In this paper, we seek to minimize the communication between the robots by utilizing a polygonal approximation of the maps of the individual robots. Recently, specialized coordination techniques have been published for certain domains. In the context of RoboCup, different coordination behaviors are used in combination with role assignment techniques [12, 19].

In the literature, several techniques are available that reduce polygons consisting of originally n vertices to similar polygons consisting of a subset of m vertices (see Heckbert and Garland [8] or Buzer [5] for comprehensive surveys). In our approach, we apply the Douglas-Peucker algorithm [6] which, according to [20], is one of the most visually effective line simplification algorithms. In the past, many improvements have been proposed for the basic Douglas-Peucker algorithm. Hershberger and Snoeyink [9] proposed an $O(n \log n)$ variant of the basic Douglas-Peucker algorithm whose time complexity is $O(nm)$. Another improvement of this algorithm that avoids self-intersecting approximations is the star-shaped Douglas-Peucker algorithm [20].

The contribution of this paper is an approach that is able to efficiently explore an unknown environment with mobile robots that only have a low bandwidth connection to exchange information. Our algorithm approximates the environment using line simplification techniques to obtain a compact geometric model which then is used to coordinate the robots. Compared to the full map, these polygonal approximations require seriously fewer memory and this way can be more efficiently communicated.

3. APPROXIMATING THE ENVIRONMENTAL MODEL

The key idea of our approach to deal with a low bandwidth communication link during exploration is to compute an approximative but compact representation of the environment and to communicate only this compact model between the robots. To obtain the compact approximation of the map, we compute a set of polygons which are extracted from a grid map of each robot. The polygons contained in a polygon map can contain either free or unknown space. Polygons that are contained inside of other polygons have a higher priority and thus overwrite the occupancy values of the outer polygons (compare Figure 1). The boundaries of a polygon can represent either free, occupied or unknown areas. These polygon maps are learned by extracting the contours of a robot's field of view and of the observed obstacles. We then apply an adapted version of the Douglas-Peucker algorithm [6] to approximate the extracted contours. In the following, the contour is also referred as the boundary $\mathcal{B}$ of the observed area.

We merge polygon maps by building a grid map in the following way. If the information given by other robots is not contradictory, a joined grid map can be constructed in a straightforward manner. If in contrast the information given by other robots is contradictory, we prefer occupied to free and free to unknown information. Note that a robot only updates its own map according to the received information in areas it has not observed on its own.

Since the field of view of each robot changes in every step, the polygonal model can get out-dated quickly. Accordingly, the polygon map needs to be updated appropriately. Since the robots only have a low bandwidth connection, it is not appropriate to transmit the whole polygon map after each update. Instead, we communicate the incremental changes of the model only. This is achieved by introducing the constraint, that points which are a part of the current polygon map model will also be part of the updated model. This constraint, of course, only holds for those points, which after the update still lie on the boundary of an obstacle or on a frontier to unknown terrain.

To update a polygon map based on sensory input, we distinguish the following cases (see Figures 2-4):

1. Vertices of the polygonal approximation of the boundary $\mathcal{B}_{t-1}$ at time step $t-1$ are no longer located on the boundary $\mathcal{B}_t$ of the observed area in the current time step t (see Figure 2),

2. the boundary of the observed area can change in a way so that all points of the approximation are still located on the boundary but the approximation becomes inadequate anyway (see Figure 3), or

3. new boundaries can arise or a boundary can split up into several boundaries which are not connected any-

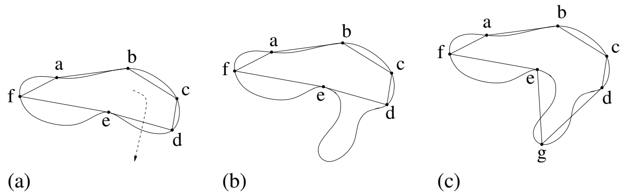

Figure 3: Case 2: All vertices of the approximation lie on the boundary $\mathcal{B}_t$ but the approximation gets inadequate anyway.

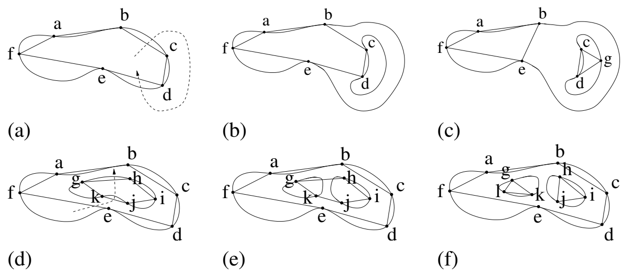

Figure 4: Case 3: A newly raised boundary (a)-(c) and a boundary which splits up into several new boundaries (d)-(f).

more and which can be contained in the original boundary (see Figure 4).

To update the polygon map, we mark all vertices which lay on the boundary of the visible area $\mathcal{B}_{t-1}$ in the previous step $t-1$ but do not lie on $\mathcal{B}_t$ anymore. After that, we iteratively discard all points that have been marked and connect their former neighbors. After this step, all points that remain part of the approximation are located on $\mathcal{B}_t$. In case points have been removed, we typically have to refine the approximation again to appropriately model the environment.

In the next step, we have to identify whether parts of the boundary $\mathcal{B}_{t-1}$ have split up into several parts (like shown in Figure 4 (a)-(c) and (d)-(f)). This is achieved by labeling the vertices of the polygon map with the number of the contour on which they are situated on $\mathcal{B}_t$. Whenever two former neighboring vertices of the polygon map now have different labels, they are disconnected by discarding the edge between these vertices. To close all polygons again the robot traces the underlying contour $\mathcal{B}_t$ of each vertex v having less than two neighbors. The process for a vertex v stops when we find another vertex w on the underlying contour that also is part of the polygon map and has the same index as v. Each pair of vertices v and w is connected and so all polygons get closed again. In case, v is the only vertex still lying on the underlying contour $\mathcal{B}_t$, a new arbitrary point w of the underlying contour is taken to close the approximation. This is also done for contours that are not yet approximated by any vertex of the new polygon map.

After this step, we can refine the new polygon map using the Douglas-Peucker algorithm by splitting up edges. Usually, the Douglas-Peucker algorithm inserts a new vertex having the biggest distance to its closest segment into its closest line segment. Since we work with sets of polygons, we would, according to the original Douglas-Peucker algorithm, split up the edge having the biggest segment-point-distance of the polygon that has the maximum segment-point-distance of all polygons.

Figure 5 shows the evolution of the approximation process for a single robot exploring an unknown environment. In this figure, white areas outside as well as gray/light blue areas inside the polygon map correspond to inappropriately modeled terrain. Such an approximation is well-suited to be transmitted via a network, because it can be updated incrementally and the individual update steps can be realized with low space complexity.

4. MULTI-ROBOT COORDINATION

Our strategy for coordinating teams of mobile robots is based on the ideas of Burgard *et al.* [4]. Their approach spreads the robots over the environment by introducing a penalty for places already visited by the robots. In contrast to this, our algorithm works in a distributed manner. Instead of using a central component which selects target locations for all robots, our approach considers the decisions so far made by other robots. Whenever a robot selects a new target location, it broadcasts its decision to all other robots. From this point in time, the other robots incorporate this information into their plans by discounting this goal according to [4]. Our approach is described in Algorithm 1. In this formulation, $V_{t'}$ refers to the cost of reaching target location t' from the current position of the robot. Our approach furthermore combines the polygonal maps of its team mates with its own world knowledge. Since data received from other robots have an approximative character, the robot only updates such parts of its own map using data received by other robots if the robot has not yet covered the corresponding area with its own sensor.

Whenever a target point t' is selected for a robot, we reduce the utility of the adjacent goals in distance d from t' according to the probability $P(d)$ that the robot's sensors will cover cells in distance d. In our approach, we approximated $P(d)$ by

$$P(d) \quad = \quad \begin{cases} 1.0 - \frac{d}{max\_range} & \text{if } d < max\_range \\ 0 & \text{otherwise} \end{cases} ,(1)$$

where max_range is the maximum range reading provided by the range sensor.

Such a coordination technique is well-suited to spread the robots over the environment and to avoid that several robots approach the same target because it is the closest to its current location. Compared to the coordination scheme

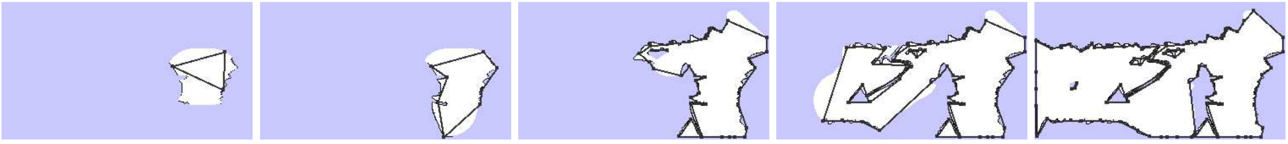

Figure 5: The approximation of the environmental model during exploration.

Algorithm 1 Goal Assignment for Coordinated, Decentralized Multi-Robot Exploration using Polygons.

Compute the union of all polygonal maps received by the other robots and combine them with the own self explored environmental data.
Compute the possible goal locations based on the frontier (points that lead to unknown areas).
Set the utility U_t of potential targets to 1.
for all Received targets from other robots **do**
  Reduce the utility of each target point t' in the visibility area according to $U_{t'} \leftarrow U_{t'} - P(||t - t'||)$.
end for
Determine the robot's target t which satisfies:
$t = \max_{t'} (U_{t'} - \beta \cdot V_{t'})$.
Broadcast the target t to all other robots.

in [4], our coordination mechanism typically leads to longer exploration times. However, it is the only possibility to use the discounting technique in a decentralized way. The centralized approach needs a much higher bandwidth, since more information needs to be exchanged.

5. EXPERIMENTS

The experiments described in this paper are designed to demonstrate the effectiveness of our environmental approximation to coordinate a team of robots using a low bandwidth network connection.

5.1. Influence of the Network Bandwidth on the Exploration Time

In this experiment, we analyzed the dependency of the quality of the environmental approximation on the overall exploration time.

The central parameter that determines approximation quality is the split rate (SR). With this parameter, we determine an upper bound for the number of edge splits per second. So when we have a SR of 0.04 the robot performs at most one edge split every 25 seconds. Edge splits are not performed when the approximation of the robot is appropriate meaning that the approximation error is equal to zero.

Figure 6 depicts how exploration time for coordinated robots decreases when the quality of the approximation (and thus the network bandwidth) increases. In these plots, we compare the performance of the original centralized coordination approach presented by Burgard *et al.* [4] to our decentralized approach. It furthermore compares our method to an adapted version of the centralized approach, which uses our decentralized coordination. It is important to note that the decentralized coordination leads to worse results than the original centralized approach, since the target assignment process is done in a decentralized way. Only a centralized coordination technique has the ability to choose that robot-target tuple that provides the (globally) highest expected utility. The decentralized method used throughout this paper is necessary to keep the communication costs low. The relative advantage of the original coordination mechanism compared to the decentralized variant increases with the number of robots. Figure 6 demonstrates that for an increasing communication link bandwidth the performance of our approach converges towards the performance of the decentralized coordination approach without polygonal approximations and bandwidth restrictions.

5.2. Comparison to Other Approaches

In the second experiment, we compared our approach to the coordination technique presented by Burgard *et al.* [4], which uses an unrestricted communication link. In their approach, the environmental model needs to be integrated by each robot of the team and the sensor information of all robots need to be transmitted to all other robots. In scenarios with a huge amount of robots, this causes a huge amount of network traffic.

Our decentralized coordination method is derived from the centralized coordination technique in [4] and was adapted to our underlying representation and to the bandwidth restrictions. Thus, the original approach with unlimited communication describes a lower bound for our algorithm. This is illustrated in Figure 6. As can be seen from this experiment, even with low bandwidth connections, which allow to transmit around 7.500 integers per second and robot, a similar coordination result can be achieved compared to the centralized coordination approach which requires a network bandwidth of about 200.000 integers per second.

5.3. Analyzing the Network Traffic

In this experiment, we analyzed the network traffic. In particular, we analyzed which kinds of data packages occurred

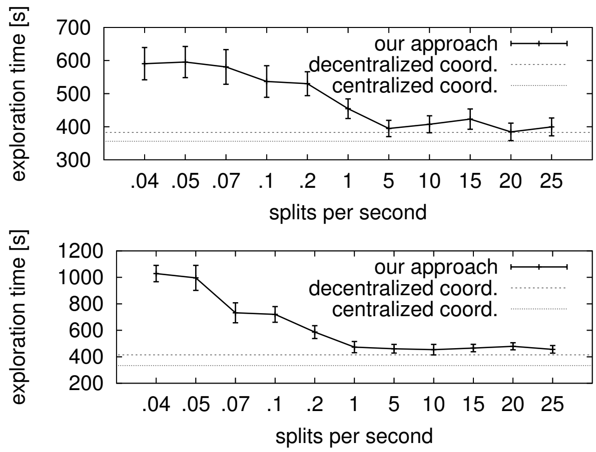

Figure 6: Exploration time for different split rates (approximation qualities) and numbers of robots. The plot in the first row shows the result for 3 robots in an unstructured environment and the second row depicts the evolution for 5 robots in a corridor environment.

SR	TPT (%)	ES (%)	B (%)	VP (%)	EP (%)
0.04	31.12	19.82	11.32	6.16	31.58
0.05	29.00	23.06	13.51	5.45	28.97
0.07	25.68	26.89	15.25	5.37	26.81
0.1	22.00	36.09	16.02	4.54	21.35
0.2	16.77	52.52	14.24	3.90	12.58
1	5.62	78.95	3.46	9.52	2.45
5	1.56	79.90	0.26	17.78	0.50
10	1.01	78.57	0.04	19.97	0.41
15	0.85	78.14	0.03	20.68	0.29
20	0.77	77.99	0.02	20.93	0.28
25	0.76	77.88	0.02	21.05	0.30

Table 1: The relative network traffic introduced by the different packages for different approximation refinement frequencies (SR).

and what was their share compared to the full amount of network traffic.

In our approach, different kinds of data packages need to be sent via the communication link. In our current implementation, we use target point transmissions (TPT) to publish chosen target locations to robots with lower priority. Edge split packages (ES) refer to the situation, in which an edge of the approximation is refined. A broken boundary package (B) is sent via the network if the boundary splits up to several small boundaries and the emerged points data package (EP) is used to describe a newly detected object (see Section 3, case 3). Finally, the vanished points package (VP) corresponds to the situation, in which a point is removed from approximation. To identify a point or edge in the approximation, we use unique IDs and each point is represented by two integer values.

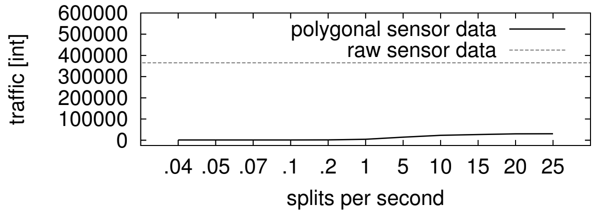

Figure 7: Comparison of network traffic amounts for the polygonal approach (for different approximation qualities) and for classical approaches transmitting raw sensor data for two robots.

Table 1 depicts the relative network traffic caused by the different data packages. It demonstrates, that in case of low approximation refinement frequencies (SR) the transmission of the goal points causes a high amount of the overall traffic (33.12%), whereas this can be neglected in case of high refinement rates.

To give a more quantitative evaluation of the caused traffic, we compared the used bandwidth of our approach to the centralized coordination approach. Using this technique, each robot transmits its full laser range sweeps to the other robots. This ensures, that all robots have the same world knowledge. Using a broadcast network, the overall network traffic grows linearly in the number of robots. Figure 7 depicts the overall network traffic caused by both approaches exploring the same environment. As can be seen, our approach clearly outperforms the other technique. Even in the case of high approximation refinement rates, our approach requires only a fragment of the network traffic compared to the centralized coordination algorithm.

5.4. Approximation Error

The last experiment in this paper is designed to illustrate the approximation error of our approach compared to a non-distributed algorithm, which integrates all sensor measurements into one central map.

The approximation error and thus the quality of an approximation directly depends on the SR whereas the higher the SR the lower is the average approximation error (see Table 2).

Figure 8 depicts the approximation error of our polygonal maps for two different approximation refinement rates. The plot in the first row depicts the error in case of a refinement rate of 10 refinement operations per second. As can be seen, the error is quite small. Whenever the robot observes a huge amount of so far unknown terrain, the approximation error increases. As can be seen from the figure, the error is typically reduced within a few steps. Compared to that, the error using a low bandwidth connection (see second row of Figure 8), which allows 0.1 refinements per second, is bigger. Furthermore, it takes much longer to correct the error

Splits per Second	Avg. Approximation Error
0.1	8.92 %
1	0.86 %
10	0.19 %

Table 2: Low communication rates lead to high approximation errors and vice versa.

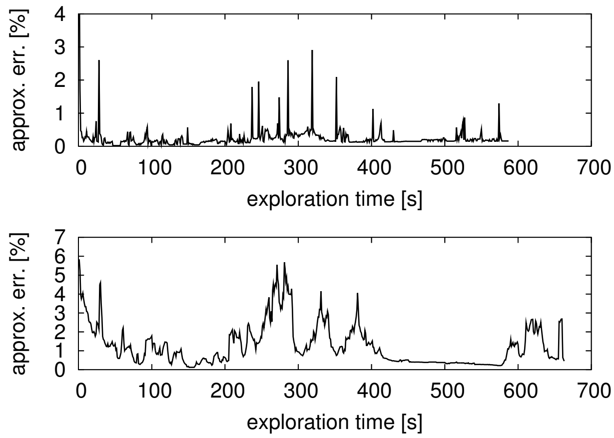

Figure 8: Approximation error during exploration for (top image) high and low (bottom image) communication bandwidth.

compared to the faster connection.

6. CONCLUSIONS

In this paper, we presented a distributed approach to multi-robot coordination for systems with reliable but limited bandwidth connections. To deal with low bandwidth networks, we approximate the maps communicated between the robots by polygonal representations. We also describe an incremental scheme for updating the polygons whenever the map has been extended. Finally, we proposed a distributed approach to assign targets to robots.

Our approach has been implemented and tested in extensive simulation runs. The results reveal that our algorithm can efficiently coordinate teams of mobile robots even under severe bandwidth restrictions. One finding is that there is no significant difference to decentralized approaches assuming unlimited bandwidth when each robot broadcasts only very little information in each step.

Acknowledgment

This work has partly been supported by the German Research Foundation (DFG) under contract number SFB/TR-8 (A3) and by the EC under contract number FP6-004250-CoSy.

7. REFERENCES

[1] D. Apostolopoulos, L. Pedersen, B. Shamah, K. Shillcutt, M.D. Wagner, and W.R.L. Whittaker. Robotic antarctic meteorite search: Outcomes. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 4174–4179, 2001.

[2] T. Balch and R.C. Arkin. Communication in reactive multiagent robotic systems. *Journal of Autonomous Robots*, 1(1):27–52, 1994.

[3] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun. Collaborative multi-robot exploration. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2000.

[4] W. Burgard, M. Moors, C. Stachniss, and F. Schneider. Coordinated multi-robot exploration. *IEEE Transactions on Robotics and Automation*, 2005. To appear.

[5] L. Buzer. A survey and a new competitive method for the planar min-# problem. *CoRR*, cs.CG/0209007, 2002.

[6] D.H. Douglas and T.K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *The Canadian Cartographer*, 10(2):112–122, 1973.

[7] B.P. Gerkey and M.J. Matarić. Sold! auction methods for multi-robot coordination. *IEEE Transactions on Robotics and Automation*, 18(5):758–768, 2002.

[8] P.S. Heckbert and M. Garland. Survey of polygonal surface simplification algorithms. Technical report, School of Computer Science, Carnegie Mellon University, Pittsburgh, May 1997.

[9] J. Hershberger and J. Snoeyink. An $O(n \log n)$ implementation of the douglas-peucker algorithm for line simplification. In *SCG '94: Proceedings of the tenth annual symposium on Computational geometry*, pages 383–384. ACM Press, 1994.

[10] A. Howard, M.J. Matarić, and S. Sukhatme. An incremental deployment algorithm for mobile robot teams. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2002.

[11] Y. Huang, Z.Cao, S. Oh, E. Kattan, and E. Hall. Automatic operation for a robot lawn mower. In *SPIE Conference on Mobile Robots*, volume 727, pages 344–354, 1986.

[12] L. Iocchi, D. Nardi, M. Piaggio, and A. Sgorbissa. Distributed coordination in heterogeneous multi-robot systems. *Journal of Autonomous Robots*, 15(2):155–168, 2003.

[13] M. Jäger and B. Nebel. Dynamic decentralized area partitioning for cooperating cleaning robots. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2002.

[14] J. Ko, B. Stewart, D. Fox, K. Konolige, and B. Limketkai. A practical, decision-theoretic approach to multi-robot mapping and exploration. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2003.

[15] H.W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1):83–97, 1955.

[16] R. Murphy. Human-robot interaction in rescue robotics. *IEEE Systems, Man and Cybernetics Part C: Applications and Reviews*, 34(2), 2004.

[17] M. Simoncelli, G. Zunino, H.I. Christensen, and K. Lange. Autonomous pool cleaning: Self localization and autonomous navigation for cleaning. *Journal of Autonomous Robots*, 9(3):261–270, 2000.

[18] S. Thrun, D. Hähnel, D. Ferguson, M. Montemerlo, R. Triebel, W. Burgard, C. Baker, Z. Omohundro, S. Thayer, and W. Whittaker. A system for volumetric robotic mapping of abandoned mines. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2003.

[19] T. Weigel, J.-S. Gutmann, M. Dietl, A. Kleiner, and B. Nebel. Cs-freiburg: Coordinating robots for successful soccer playing. *IEEE Transactions on Robotics and Automation*, 18(5):685–699, 2002.

[20] S.-T. Wu and M.R. Gonzales Márquez. A non-self-intersection douglas-peucker algorithm. In *SIBGRAPI*, pages 60–66, 2003.

[21] B. Yamauchi. Frontier-based exploration using multiple robots. In *Proc. of the Second Int. Conf. on Autonomous Agents*, 1998.

[22] R. Zlot, A.T. Stenz, M.B. Dias, and S. Thayer. Multi-robot exploration controlled by a market economy. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2002.