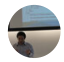
Multi-robot area coverage with limited visibility.

Conference Paper · January 2010

DOI: 10.1145/1838206.1838451 · Source: DBLP

4 authors:

Pooyan Fazli
Carnegie Mellon University



Alireza Davoodi
University of British Columbia

Philippe Pasquier
Simon Fraser University

Alan Keith Mackworth
University of British Columbia

Multi-Robot Area Coverage with Limited Visibility (Extended Abstract)

Pooyan Fazli[1]★, Alireza Davoodi[2]★, Philippe Pasquier[2] and Alan K. Mackworth[1]
[1]Department of Computer Science, University of British Columbia
[2]School of Interactive Arts and Technology, Simon Fraser University
{pooyanf, mack}@cs.ubc.ca, {ada48, pasquier}@sfu.ca

ABSTRACT

We address the problem of multi-robot area coverage and present a new approach in the case where the map of the area and its static obstacles are known and the robots have a limited visibility range. The proposed method starts by locating a set of static guards on the map of the target area and then builds a graph called *Reduced-CDT*, a new environment representation method based on the *Constrained Delaunay Triangulation (CDT)*. *Multi-Prim's* is used to decompose the resultant graph into a forest of *partial spanning trees (PSTs)*. Each *PST* is modified through a mechanism called *Constrained Spanning Tour (CST)* to build a cycle which is then assigned to a covering robot. Subsequently, robots start navigating the cycles and consequently cover the whole area. The proposed approach is complete provided that at least one of the robots operates correctly.

Categories and Subject Descriptors

I.2.9 [**Artificial Intelligence**]: Robotics – Commercial robots and application; I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence – Multiagent systems

General Terms

Algorithms, Performance, Experimentation

Keywords

Multi-Robot Systems, Teamwork, Coordination, Area Coverage, Reduced-CDT, Multi-Prim's, Constrained Spanning Tour

1. INTRODUCTION

Multi-robot area coverage is receiving considerable attention due to its applicability in different scenarios such as search and rescue operations, planetary exploration, intruder detection, environment monitoring, floor cleaning and so on. In this task a team of robots is cooperatively trying to observe or sweep an entire area, possibly containing obstacles, with their sensors or actuators. The goal is to build an efficient path for each robot which jointly ensure that every single point in the environment can be seen or swept by at least one of the robots while performing the task.

There is confusion in the literature regarding the terms *Coverage* and *Exploration*. To clarify the problem definition, we note that in exploration, there exists an unknown environment and a team of robots is trying to build a map of the area together [2]. In a coverage problem, the map of the environment may be known or unknown and the goal of the team is to jointly observe/sweep the whole area with their sensors or physical actuators. Building a map of the environment is not the ultimate aim of the coverage mission [4].

2. MULTI-ROBOT AREA COVERAGE WITH LIMITED VISIBILITY

We present a cooperative approach to covering a known environment using an arbitrary number of robots. The robots are assumed to have 360° field of view and a predefined circular visibility range. Our coverage method is composed of four main steps: First, it determines the locations of static guards required to visually cover a given 2D environment, considering the limited visibility range of the robots' cameras. Then, it builds the *Reduced-CDT*, a graph-based representation of the environment. An algorithm called *Multi-Prim's* is introduced to partition the graph and construct a forest consisting of as many *partial spanning trees* as there are covering robots. Afterward, a new method called *Constrained Spanning Tour (CST)* is used to build a cycle on each resultant tree of the forest, and finally, the cycles are allocated individually to the robots.

2.1 Locating Guards

In our problem definition, we assume the robots are equipped with panoramic cameras with a 360° field of view. However, the cameras' visibility range is limited. Our approach uses a variant of the algorithm proposed by Kazazakis *et al.* [6] and Seidel [7] to locate a minimal number of guards required to visually cover an entire area. These static guards are control points that can jointly cover the whole environment while satisfying the visibility constraint of the robots. To this end, the proposed approach decomposes the initial target area, a 2D, simple, possibly non-convex polygon with static obstacles, into a collection of convex polygons (Trapezoidal Decomposition). Then, a divide and conquer method is applied to successively divide each of the resultant convex polygons into smaller convex sub-polygons until each of them can be visually covered by one guard.

★The authors wish it to be known that, in their opinion, the first two authors should be regarded as joint First Authors

Cite as: Multi-Robot Area Coverage with Limited Visibility (Extended Abstract), Pooyan Fazli, Alireza Davoodi, Philippe Pasquier, Alan K. Mackworth, *Proc. of 9th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2010)*, van der Hoek, Kaminka, Lespérance, Luck and Sen (eds.), May, 10–14, 2010, Toronto, Canada, pp. XXX-XXX.

2.2 Environment Representation

We investigate a graph structure for environment representation based on the *Constrained Delaunay Triangulation (CDT)* [3]. Given the set of obstacles and their corresponding endpoints, the algorithm first uses the method explained in the previous section to create the set of static guards. The *CDT* is then built on the obstacles and the computed guards.

2.2.1 Graph Reduction

The aim of graph reduction is to improve efficiency by minimizing the average or total time taken for the robots to traverse the graph. The input of the algorithm is the *CDT* made on the map of the area.

The method starts by using the *Floyd-Warshall* algorithm to find the set *MD* of minimum distances, and the set *SP* of shortest paths between any pair of vertices of the input graph.

The minimum value of all the minimum distances in *MD* is then selected provided that both the endpoints of the corresponding shortest path in *SP* belong to the set of static guards computed in the previous section. The chosen path, including all its vertices and edges, forms the initial component called *Connected Component*.

Next, among all the guards that have not yet been added to the component, the algorithm finds the closest guard to the current component, merging the corresponding shortest path with it. Following the same process, the algorithm keeps expanding the *Connected Component* until there are no more guards to be added. The resultant *Connected Component* is the final reduced graph called *Reduced-CDT*.

2.3 Multi-Prim's Algorithm

The *Multi-Prim's* algorithm [5] extends the *Prim's* algorithm [1] used to build the *minimum spanning tree* of a weighted graph. This extension is motivated by the fact that multiple robots are involved in the environment to accomplish the task. The proposed approach has a weighted graph (*i.e. Reduced-CDT*) as an input and outputs a forest consisting of as many *partial spanning trees* as there are covering robots. These trees are created incrementally from the initial location of the robots.

The *Multi-Prim's* algorithm starts by determining the starting points. A corresponding starting point for a robot is a visible vertex of the *Reduced-CDT* closest to the robot. In some situations the algorithm might lead to the same starting points for different robots. Subsequently, robots try to sequentially expand their own trees (one edge at a time) using *Prim's* algorithm until all the vertices of the reduced graph are visited at least once. The vertices are visited in a way that satisfies the following three constraints: (1) Find the nearest adjacent vertex, adding it and the corresponding edge to the tree provided that it does not create a cycle. (2) Avoid adding a vertex which has already been visited by the other robots, unless there is no other unvisited adjacent vertex. (3) Terminate the algorithm when all the vertices of the graph have been visited by at least one robot.

2.4 Constrained Spanning Tour

The next step is to construct a cycle on each partial spanning tree resulted from *Multi-Prim's* algorithm. To this end, we introduce an algorithm called *Constrained Spanning Tour (CST)* which is an improved variant of the *Double-Minimum Spanning Tree (Double-MST)* algorithm.

Double-MST takes a tree as an input and returns a cycle whose length is twice the length of the tree and each vertex is visited exactly two times. A revision can be made to the algorithm in order to form a possibly shorter cycle. Starting from an arbitrary initial point, *CST* traverses the vertices in the same way as the *Double-MST* algorithm does, but whenever it reaches a vertex visited before, discards it, proceeding to the next vertex along the cycle to find an unvisited one, making a shortcut edge to it. However, because of the existing obstacles in the environment, *CST* uses only the edges of the original graph, *CDT*, as shortcuts. This algorithm uses backtracking to find out the best shortcut. If the shortcut does not belong to the original graph, the next best shortcut will be considered. *CST* traverses the tree to return back to the starting point.

Finally, the robots start navigating the cycles which consequently results in full coverage of the target area. *CST* has the benefit that it returns the robots to their initial locations, facilitating tasks like garbage collection and storage.

3. CONCLUSION AND FUTURE WORK

This paper investigated the multi-robot area coverage problem and presented a new approach for covering a known polygonal area cluttered with static obstacles. The robots' cameras are assumed to have a limited visibility range. The proposed approach can be used in other applications such as border inspection instead of area coverage. It is also guaranteed to be complete provided that at least one robot operates correctly. There are numerous challenging future research directions for this problem. Some are as follows: **Heterogeneity:** Dealing with differing movement or sensing capabilities of the robots. **Open Systems:** Adding/removing the robots to/from the team in the course of the coverage mission. **Priority:** observing or covering parts of the target area sooner than others due to different priorities. **Communication:** handling cases with limited range of communication, meaning a message sent by a robot is transmitted only to robots within a certain distance from that robot. **Dynamic Environments:** handling dynamic obstacles or an environment changing in shape or size.

4. REFERENCES

[1] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows: Theory, Algorithms, and Applications.* Prentice Hall, February 1993.

[2] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun. Collaborative multi-robot exploration. In *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA '00.*, volume 1, pages 476–481, 2000.

[3] L. P. Chew. Constrained delaunay triangulations. In *Proceedings of the Symposium on Computational Geometry*, pages 215–222, 1987.

[4] H. Choset. Coverage for robotics - a survey of recent results. In *Annals of Mathematics and Artificial Intelligence*, volume 31, pages 113 – 126, 2001.

[5] A. Davoodi, P. Fazli, P. Pasquier, and A. K. Mackworth. On multi-robot area coverage. In *Proceedings of the 7th Japan Conference on Computational Geometry and Graphs, JCCGG09*, pages 75–76, November 2009.

[6] G. D. Kazazakis and A. A. Argyros. Fast positioning of limited visibility guards for inspection of 2D workspaces. In *Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2843–2848, October 2002.

[7] R. Seidel. A simple and fast incremental randomized algorithm for computing trapezoidal decompositions and for triangulating polygons. *Computational Geometry: Theory and Applications*, 1(1):51–64, 1991.