Proceedings of IEEE International Conference on Applied System Innovation 2018
IEEE ICASI 2018- Meen, Prior & Lam (Eds)

Task coordination for Multiple Mobile Robots considering Semantic and Topological Information

Ankit A. Ravankar∗, Abhijeet Ravankar∗†, Chao-Chung Peng‡, Yukinori Kobayashi∗, and Takanori Emaru∗

· Research Faculty of Engineering, Hokkaido University, Japan
†Kitami Institute of Technology, Kitami, Japan
‡Department of Aeronautics and Astronautics, National Cheng Kung University, Tainan, Taiwan

Abstract

This paper presents task coordination scheme for multiple mobile robots in indoor environment. Multi-robot systems are gaining a lot of popularity for different applications in indoor and outdoor environments and for executing various tasks in different scenarios. The advantage with multi-robot systems is efficiency by which a task can be completed as compared to single robot system thus allowing more autonomy to the task at hand. In this paper, we present novel techniques for multi-robot system operating in indoor office like environment using semantic mapping. A mapping and navigation algorithm is discussed that uses topological and metric maps. We demonstrate through simulations and experiments, the coordination strategy and discuss the results.

Key words: Multi-robot system, semantic mapping, SLAM, exploration, task coordination.

Introduction

In the last decade, there has been a rapid progress in the usage of mobile robots for different applications in industries and at home. They are continuously being employed for tasks such as cleaning, surveillance and patrolling, warehouse keeping, at hospitals, for search and rescue operations and many more applications. Out of all these, home and office environments constitute the main location where robots are meant to be useful to humans by assisting them in completing simple or complex tasks. Yet, present robot systems are not capable to complete the expectation in terms of intelligence, knowledge and skills. Sensor limitations, clutterness in the environment and communication failure limits the performance of such robots. While single robot systems have been successfully deployed in different applications, multi-robot systems or MRS are recently becoming very popular. They are designed to perform collective behavior towards common or different goals. Through this collective behavior, tasks that are difficult for a single robot to achieve becomes feasible for MRS. There are several benefits of using MRS, some of them are (1) Wide area coverage, (2) Task distribution (3) Task parallelism, (4) Continuous operation, (5) Improved task efficiency, (6) Cooperative task execution and, (7) Fault tolerance [6]. Although MRS have many advantages and practical importance they too suffer from challenges such as collision avoidance, networking, task and priority planning, conflict resolution and inter-robot communication. The type of environment and number of robots also affect the task at hand. Navigation and mapping in indoor environments is still a challenging problem considering that the robots operate in large environments with similar looking features that adds to the complexity. At the same time, most traditional methods for navigation and mapping only considers 2D sensor data when mapping the environment. This can be representing the environment as a line map or grid maps. However, largely the robots operate in 3D environments and it is important for them to have complete perception of the environment they are operating in. Very recently advances in sensor fusion and computer vision have resulted in 3D mapping of environment. Although, such methods are mostly applied on small scale areas and require large computation and memory for storing the 3D information from the scene. Localization is another big challenge in MRS. Sensor uncertainty and size of the environment results in large errors that mostly produces loop closure failure or distorted maps. Many researchers have only focused on the SLAM or the simultaneous localization and mapping problem considering the challenges it poses to the MRS and its importance in truly autonomous applications [10]. Coordination between the team of robots in MRS is critical step in achieving success. Largely, they are task allocation and task coordination. While the former represents the allocation of different "tasks" to the team of robots and which robot takes what job (or sub goals) based on its configuration, task coordination implies how the task will be handled by the MRS. Our goal in this research work is to combine both problems based on semantic map information from the scene. By semantic mapping, we mean to denote the process that allows the robot to enrich the map for navigation using semantics (i.e. knowledge about the environment). This will allow the robots to understand the scene they are operating in and make intelligent judgement based on the knowledge they acquire during navigation. Thus, allowing online scheduling of tasks

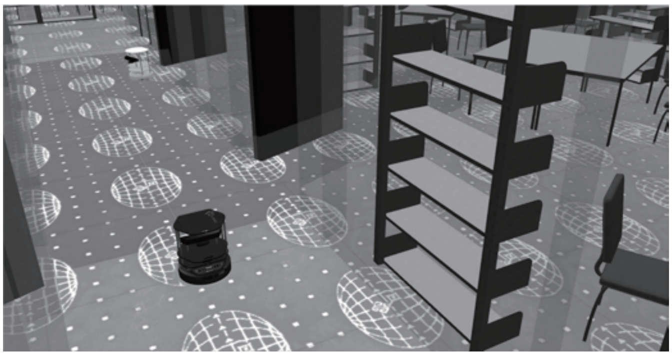

Fig. 1: Multiple robots in office like environment

ISBN 978-1-5386-4342-6

Authorized licensed use limited to: University of Ottawa. Downloaded on January 08,2022 at 20:56:43 UTC from IEEE Xplore. Restrictions apply.

and clear distribution of jobs in well-structured environments. In this paper, to solve this problem, we first present our hybrid mapping technique that uses metric and topological mapping to construct a layered map of the environment. Based on the map, a team of robot is deployed to complete certain tasks autonomously. The hybrid maps allow the robot to have semantic information from the environment and cluster areas of similar regions for better exploration and coordination. From the semantic information, the robots can autonomously share and prioritize tasks between them and capture areas of the map based on the clustered regions.

Rest of the paper is as follows: Section 2 gives a summary of the related works in the area, Section 3 presents the mapping system. Section 4 presents algorithm for task coordination and scheduling with results, and Section 5 concludes the paper.

Related Works

Works in robot mapping and localization or SLAM has been extensively done in both single robot systems and MRS. For single robot SLAM, an overview of some of the recent advances in SLAM can be found in literature [1, 12]. For multi-robot systems numerous works have been presented in the past that focuses on different aspects of coordination and task planning for MRS systems. For e.g. Recent works such as [7–9], presents a multi-robot system that are based on leader-follower scenarios, symbiotic navigation and bio-inspired pheromone trailing respectively. [3, 13], shows a survey on coordination strategies for multi-robot exploration and mapping. Whereas work presented in [2], presents MRS systems using techniques from various fields such as operations research, economics, scheduling, network flows, and combinatorial optimization.

Proposed Method

The proposed method for the MRS is presented here. We first briefly present our hybrid mapping system followed by the task scheduling algorithm.

A. Hybrid Mapping System

Our proposed algorithm uses semantic mapping for the robots to navigate in complex environments. Figure 2 represents the hybrid mapping system. It consists of a hybrid mapping subsystem and a navigation subsystem together making the SLAM system. The robot equipped with a planer LIDAR sensor provides wheel odometry and laser scan data to the local metric map. In this research we considered the grid maps for the local metric map that is build using the particle SLAM algorithm as described in [12]. A topological map is built on top of the metric map that consists of nodes and edges that corresponds to local robot poses and laser scans. The topological map is maintained globally storing all information on a global scale and is periodically optimized to find loop closure whereas the metric map (grid map) is maintained locally. This has an added advantage as maintaining only a part of the local map allows to save enough information for the robot to do local navigation. Unlike traditional methods, where a complete grid map is stored and can take significant memory

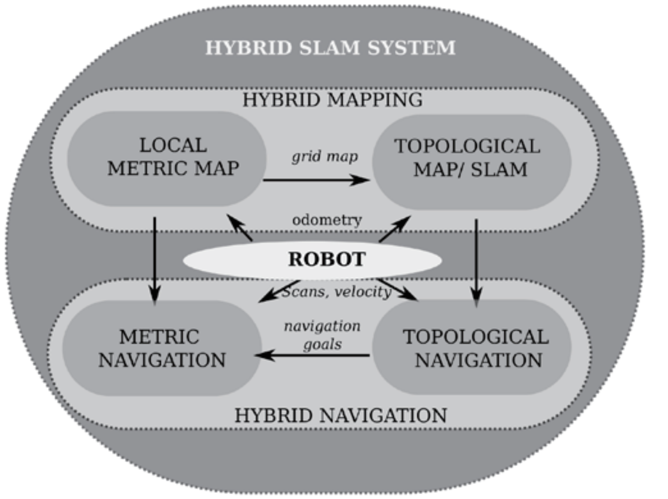

Fig. 2: Proposed hybrid semantic mapping system

in cases when the environment is large, the proposed method allows only to store fraction of the memory. All relevant data from the robot's sensor are being stored globally in the topological graph map that requires much less memory storage as compared to the grid maps. For topological maps we used the system described in [4, 5, 11]. When the robot is given a topological node ID for navigation, the topological navigation system uses the global topological map for navigation and finds the shortest path. The topological map is converted into a series of metric goals that is passed on to the metric navigation system. Overall the two subsystems (topological and metric) works in tandem to each other thus making the overall hybrid SLAM system. Our proposed system is very much inspired by human like navigation where only relevant information are stored and only local knowledge is used for navigation. This allows the system to send goals to specific node IDs and faster route searching. The node IDs can store semantic information about the environment and are used for more human friendly navigation goal commands such as: go to the kitchen. Once the map is completed, semantic information from the scene is used to cluster the areas into labelled regions. This is primarily done using the onboard RGBD camera on the robots. A dictionary of all the items in the map is already available to the user. The objects in the scene are matched with incoming RGBD data and a key-value pair is assigned with that region (e.g. Dining table should be the dining room etc.). This semantic map is available to each robot and localization in the map is done using the sensor data (Lidar, RGBD camera, Node Ids, object detection). The robots navigate in the map using node information.

Multi-Robot Task Scheduler

Once the semantic mapping has been completed and the nodes with semantic information have been clustered, we get the total number of 'm' service locations L_j, $j \in 1, 2, ..., m$. The number of robots ('n') available are known beforehand and indexed as R_i, $i \in 1, 2, \dots, n$. The task of allocating 'n' robots to 'm' service locations is shown in Algorithm 1. Such a scheduler is important due to the diversity of the task requirements and dynamics of the availability of robots. Some service locations are large in area and it might efficiently be served well by

ISBN 978-1-5386-4342-6

Authorized licensed use limited to: University of Ottawa. Downloaded on January 08,2022 at 20:56:43 UTC from IEEE Xplore. Restrictions apply.

multiple robots operating in that location simultaneously. In

such cases, if there are extra robots available, they must be allotted to those areas. However, this should be done after other locations have been assigned the robots. Some locations in the map might have high priority. The specification of the robots themselves is another important factor. In this work, it is assumed that all the robots have the same specifications. However, diversity in the nature of service locations are considered assuming that some service locations are larger than others and require multiple robots for efficient service. Depending on the number of locations and robots available, there are three scenarios which are explained below:

1) No. of robots are same as service locations: The simplest scenario is when the number of service robots are same as that of service locations, i.e. n = m. In this case, the algorithm performs a one-to-one mapping of robot and service locations. [Algorithm1 : Lines 1 to 5]
2) No. of robots are more than service locations: The scenario when n > m is favorable as there are more robots to serve the areas and parallel task execution is possible. In this case, the algorithm works by first sorting the locations (Lj) in descending order of their areas. Thus, location L1 has the largest area and location L_m has the smallest area. The robots $R_1, R_2, \cdots, R_m$ are then allotted to locations $L_1, L_2, \cdots, L_m$, respectively. There are $n - m$ extra robots which are then allotted again to locations $L_1, L_2, , L_{n-m}$. Since, the location has been sorted in descending order, the largest area services are prioritized to get multiple robots for parallel execution. In line 12 of Algorithm 1, mod represents the modulo operator. [Algorithm1 : Lines 6 to 17].
3) More service locations than robots: If there are more locations to serve than the robots at disposal, then the algorithm works by checking if an idle robot and an unfinished location is available. It then allots the idle robot to the unfinished location. This is continued until there are no unfinished tasks at hand. [Algorithm1: Lines 18 to 24]

In all the cases, once a service location has been acquired by a robot, that location is locked by the robot. Only unlocked locations can be accessed by the robots. This is to ensure that once a location has been locked by a robot, no other robot will go to that location, and the possibility of unnecessary task duplication (e.g. cleaning a room which has already been cleaned) is avoided. While Algorithm 1 only considered the area of the service locations, other parameters can be used to prioritize the scheduler. For such cases, the strategy of sorting the service locations by the determining factor (e.g. area or other) will work similarly. It is also possible to have multiple factors for prioritization which could be weighted by an objective function and locations can be prioritized accordingly.

A. Results

We experimented in a simulated environment considering two robots in an office like scenario with different rooms. The

Algorithm 1: Multi-Robot Task Scheduler

```
   Data: Robots : R_i, i ∈ (1,2,··· ,n)
   Data: Locations : L_j, j ∈ (1,2,··· ,m)
   ···
   // No. of robots same as no. of locations; n=m
1  if n = m then
2      while idx = 1 : n do
3          allocate_task(R_idx, L_idx)
4          lock(L_idx)
5      return;
   // No. of robots are more than locations; n>m
6  if n > m then
7      n_extra_robots ← n − m
       // Sort locations by area.
       // L1 has largest area. Lm has smallest area.
8      sorted_locations ← sort_by_area(L_i)
       // First one-to-one robot:location allocation
9      while idx = 1 : m do
10         allocate_task(R_idx, sorted_locatons_idx)
       // Then extra robot:location allocation
11     while idx = 1 : n_extra_robots do
12         if idx mod m ! = 0 then
13             loc_idx ← idx modulo m
14         else
15             loc_idx ← m
16         allocate_task(R_idx, sorted_locatons_loc_idx)
17     return;
   // No. of robots are less than locations; n<m
18 if n < m then
       //Until unfinished tasks exists
19     while unfinished_task() = True do
           //Get first idle robot's ID
20         free_robot_id ← get_idle_robot(R)
           //Get first unfinished location's ID
21         location_id ← get_unfinished_location(L)
           //Allocate Idle-Robot:Unfinished-Location
22         allocate_task(R_free_robot_id, L_location_id)
23         lock(L_location_id))
24     return;
```

robots have the same specifications with similar sensors. We present case 3 where there are more locations to serve than the number of robots available. In this case n=2 and L=10 (9 rooms and 1 corridor). Note that, all areas in the map are already segmented based on our semantic mapping algorithm and each robot has this map of the environment. The robots task could be anything depending on the application. For our simulation, we programmed the robots to patrol the complete area. The semantic map with each labelled region is given in Figure 3. It shows 9 different regions (L1-L9) and corridor C. The robots start patrolling from the similar location and execute the task. Based on our proposed algorithm, the first robot takes the nearest region and captures it, while the second robot moves to the next location and so on. The area segmentation is done based on our door detection algorithm [14]. The robots also maintain the dictionary of objects in the regions and are able to localize themselves using the sensors. An advantage of the semantic map for the task coordination is that the robots can be assigned to patrol specific regions of the map based on the semantic data. For e.g. Robot 1 can be assigned to only patrol the corridor area while robot 2 to only patrol any of the regions from L1 to L9. Hence, the robots need not have to be specifically programmed by giving map coordinates and the task can be automated. From our experiments we found that it took 968 seconds for the single robot to patrol the whole map, while for the two-robot system it took 574 seconds. The robots could patrol the map autonomously and returned to the original location after finishing patrolling. From the simulation

ISBN 978-1-5386-4342-6

Authorized licensed use limited to: University of Ottawa. Downloaded on January 08,2022 at 20:56:43 UTC from IEEE Xplore. Restrictions apply.

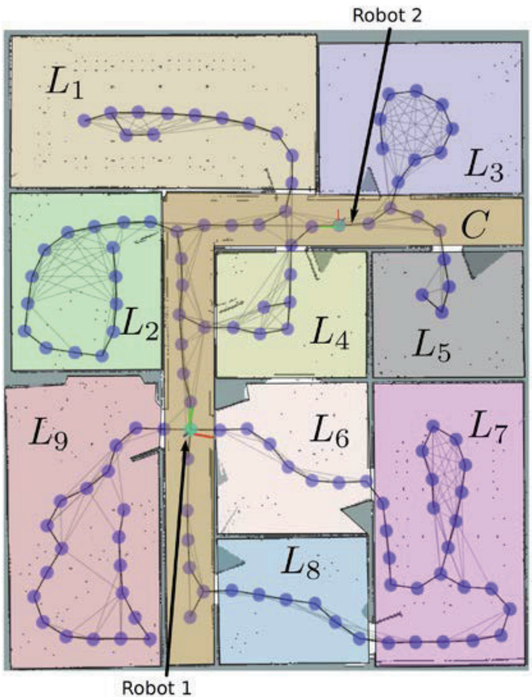

Fig. 3: Semantic map for MRS task scheduling and coordination.

experiments, we confirmed the effectiveness of our proposed system.

Conclusion and Future works

In this paper, we presented a novel multi-robot task co-ordination algorithm using semantic mapping. Our system benefits from the hybrid mapping system that consists of the metric and topological maps. Using the hybrid maps, the multi-robot system were able to autonomously distribute and coordinate tasks autonomously. With semantic information from the environment, area segmentation and area specific tasks can be easily coordinated to robot teams. In future we want to test our proposed algorithm in real world scenarios

References

[1] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scara- muzza, J. Neira, I. Reid, and J. J. Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. IEEE Transactions on Robotics, 32(6):1309–1332, 2016.

[2] B. P. Gerkey and M. J. Matarić. A formal analysis and taxonomy of task allocation in multi-robot systems. *The International Journal of Robotics Research*, 23(9):939– 954, 2004.

[3] C. Nieto-Granda, J. G. Rogers III, and H. I. Christensen. Coordination strategies for multi-robot exploration and mapping. *The International Journal of Robotics Re- search*, 33(4):519–533, 2014.

[4] A. Pronobis and P. Jensfelt. Large-scale semantic map- ping and reasoning with heterogeneous modalities. In *Robotics and Automation (ICRA), 2012 IEEE Interna- tional Conference on*, pages 3515–3522. IEEE, 2012.

[5] A. Pronobis, O. Martinez Mozos, B. Caputo, and P. Jens- felt. Multi-modal semantic place classification. *The International Journal of Robotics Research*, 29(2-3):298– 320, 2010.

[6] A. Ravankar. Novel algorithms for multi-robot path planning, task coordination, mapping and localization in indoor environment, phd thesis, hokkaido university, japan. 2017.

[7] A. Ravankar, A. A. Ravankar, Y. Kobayashi, and T. Emaru. On a bio-inspired hybrid pheromone signalling for efficient map exploration of multiple mobile service robots. *Artificial Life and Robotics*, 21(2):221–231, Jun 2016.

[8] A. Ravankar, A. A. Ravankar, Y. Kobayashi, and T. Emaru. Hitchhiking robots: A collaborative approach for efficient multi-robot navigation in indoor environ- ments. *Sensors*, 17(8):1878, 2017.

[9] A. Ravankar, A. A. Ravankar, Y. Kobayashi, and T. Emaru. Symbiotic navigation in multi-robot systems with remote obstacle knowledge sharing. *Sensors*, 17(7): 1581, 2017.

[10] A. A. Ravankar, Y. Hoshino, A. Ravankar, L. Jixin, T. Emaru, and Y. Kobayashi. Algorithms and a frame- work for indoor robot mapping in a noisy environment using clustering in spatial and hough domains. *Interna- tional Journal of Advanced Robotic Systems*, 12(3):27, 2015.

[11] A. A. Ravankar, A. Ravankar, T. Emaru, and Y. Kobayashi. A hybrid topological mapping and navigation method for large area robot mapping. In *Society of Instrument and Control Engineers of Japan (SICE), 2017 56th Annual Conference of the*, pages 1104–1107. IEEE, 2017.

[12] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005. ISBN 0262201623.

[13] Z. Yan, N. Jouandeau, and A. A. Cherif. A sur- vey and analysis of multi-robot coordination. *In- ternational Journal of Advanced Robotic Systems*, 10(12):399, 2013. doi: 10.5772/57313.

[14] A. A. Ravankar, A. Ravankar, T. Emaru and Y. Kobayashi, "A hybrid topological mapping and navigation method for large area robot mapping," 2017 56th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE), Kanazawa, 2017, pp. 1104-1107. doi: 10.23919/SICE.2017.8105770

Authorized licensed use limited to: University of Ottawa. Downloaded on January 08,2022 at 20:56:43 UTC from IEEE Xplore. Restrictions apply.