Proceedings of the 2012 IEEE International Conference on
Cyber Technology in Automation, Control and Intelligent Systems
May 27-31, 2012, Bangkok, Thailand

Decentralized Waypoint-based Multi-robot Coordination

Nicolas Jouandeau
Advanced Computing Laboratory of Saint-Denis (LIASD)
Paris 8 University
93526 Saint-Denis, France
Email: n@ai.univ-paris8.fr

Zhi Yan
Advanced Computing Laboratory of Saint-Denis (LIASD)
Paris 8 University
93526 Saint-Denis, France
Email: yz@ai.univ-paris8.fr

***Abstract*—This paper describes a decentralized approach based on Alternative Waypoint Generating (AWG) for planning separate kinematic paths to multiple robots to alleviate the waiting situation problems. The basic thought of this approach is straightforward: If a robot request a target waypoint that has already been assigned to another robot, then this robot will consider this waypoint as an obstacle and attempt to get an alternative one around it. The proposed strategy has been implemented and evaluated in simulation. The experimental results demonstrated that the waiting situation problems have been decreased and the overall system performance has been improved.**

I. Introduction

Multi-robot motion planning is one of the most challenging tasks in multi-robot systems. A very common problem in the motion planning for a group of mobile robots is when multiple robots move to the same waypoint, causing collision, congestion and deadlock, then may reduce system performance. We defined this kind of dynamic standstill of system caused by waypoint mutual exclusion as the waiting situation problem. Because, for better or worse, a robot should wait until the other one pass first, or all robots should wait for planner to replan their trajectories.

Therefore, we need some sort of coordination mechanism to harmonize the motion of robots, so as to minimize the waiting situation problem. In this paper, we present a decentralized approach based on Alternative Waypoint Generating (AWG) for planning separate kinematic paths to multiple robots. Whenever a robot request a target waypoint that has been assigned to another robot, then the former will consider this waypoint as an obstacle and try to find an alternative one around it.

Our previous work [1] focused on the problem of multi-robot exploration by using separate topological graph based on sampling environment map iteratively. The proposed investigation is related to the approach described in this paper, however, the former is based on centralized mechanism, which needs a central agent to process the map and assign exploration target points to robots, if this agent fails, the whole system will fail. In contrast, the decentralized mechanism does not need the central agent, where robots use locally observable informations to make their plans. This mechanism has a good adaptability and strong robustness, although the solutions it got are often sub-optimal.

The remainder of the paper is organized as follows: Section II describes an overview of some related works; followed by the discussion of the problem of waiting situation, described in Section III; subsequently, Section IV describes our decentralized waypoint-based method; Section V describes the experimental results obtained with our method; the paper is concluded in Section VI.

II. Related Work

Jäger and Nebel [3] described a decentralized method for coordinating the independently planned trajectories of multiple mobile robots to avoid collisions and deadlocks among them. Their idea is: for the collision, whenever the distance between two robots drops below a certain value, they exchange information about their planned trajectories and determine whether they are in danger of a collision. If a possible collision is detected, they monitor their movements and, if necessary, insert idle times between certain segments of their trajectories in order to avoid the collision. For the deadlock, whenever a deadlock is detected, the trajectory planners of each of the involved robots are successively asked to plan an alternative trajectory until the deadlock is resolved. Moors *et al.* [4] presented a graph-based algorithm for coordinate multi-robot motion planning in 2D indoor environments. The scenario of this research is multi-robot indoor surveillance. The proposed approach takes the limitations and uncertainties of sensors into account, and generates the coordinated motion plan for multiple robots by using A* search algorithm. The authors also introduced a framework based on realistic probabilistic sensor models and worst case assumptions on the intruder's motions in order to compare different approaches and evaluate the coordination performance of the proposed approach. Marcolino and Chaimowicz [5] proposed a decentralized coordination algorithm to control the traffic of a swarm of robots, avoiding congestion situations when large groups of robots move in opposite directions. The proposed algorithm allows the robots to perceive the possibility of collision and warn their teammates through local sensing and communication, then the group changes its trajectories to avoid congestion. They [6] also proposed another coordination algorithm for the control of traffic when robots try to reach the same target, where robots control their actions using a probabilistic

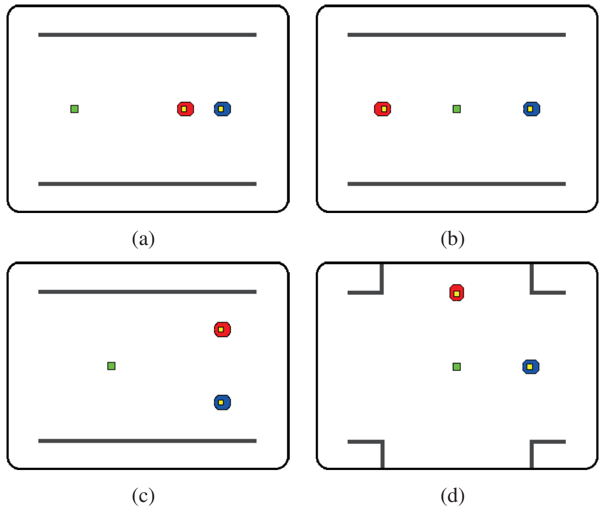

Fig. 1. Four typical situations of waypoint mutual exclusion. The green small cube represents the waypoint.

finite state machine and always rely on local sensing and communication to coordinate themselves. Besides, there are some other approaches developed with various policies [7] [8].

III. Problematic

Generally speaking, if multiple requests targeting the same resource arrive simultaneously, *resource conflict* will occur. This issue has been studied in many forms. A well known case of a general resource conflict problem is the *mutual exclusion* problem [9] in distributed computing systems or multi-access networks, where only a single resource is available. In multiple mobile robot systems (MMRS), resource conflict arises when multiple robots need to share space, manipulable objects or communication media. We focus in this paper on the space sharing problem, which is studied through multi-robot motion planning and the collision, congestion and deadlock avoidance problems.

The problem of resource conflict is usually involved in the form of waiting situation caused by waypoint mutual exclusion over the multi-robot motion planning. Figure 1 depicts four typical situations of waypoint mutual exclusion. Two robots move to the same waypoint simultaneously: In Figure 1(a), both the red and blue robot move from the right towards the left one after another (same path same direction); In Figure 1(b), the red robot moves from the left towards the right and the blue robot moves from the right towards the left (same path different direction); In Figure 1(c), both the red and blue robot move from the right towards the left side by side (different path same direction); In Figure 1(d), the red robot moves from the top towards the bottom and the blue robot moves from the right towards the left (different path different direction). Consequently, since the waypoint can be assigned only to one robot at a time, then the mutual exclusion of the waypoint happens.

IV. Our Method

Generally, there are two ways to deal with the waypoint mutual exclusion. One is to let robots pass the waypoint one by one (OBO) [2]. The weakness of this strategy is that one robot must wait for another robot to pass. Another way is to replan the local path in real time for each robot by using some goal seeking obstacle avoidance algorithms such as Vector Field Histogram (VFH+) [10] or Nearness Diagram (ND) Navigation [11]. The weakness of this strategy is that robots need some time to replan their new trajectory. Consequently, the two ways both extend the time of the motion planning and limit the system efficiency.

If we can plan separate kinematic paths for multiple robots, then the waiting situation caused by waypoint mutual exclusion will be significantly reduced. Our method is summarised in Algorithm 1.

Algorithm 1 Alternative Waypoint Generating (AWG)

Require: An occupied waypoint
Ensure: An alternative waypoint
1: **if** the requested waypoint p is occupied **then**
2: Find the nearest obstacle point o on the right side of p
3: Calculate the distance d between o and p: $d = |po|$
4: **if** $d < 2(r + e)$, where r the radius of the minimum circle to cover the robot with centering at the rotation center of robot, and e a positive number **then**
5: **return false**
6: **else**
7: Find the free point q between o and p: $|pq| = r + e$
8: **if** there is no path between q and p', where p' the previous waypoint **then**
9: **return false**
10: **else**
11: **return** alternative waypoint q
12: **end if**
13: **end if**
14: **end if**

When a robot requests a waypoint and this waypoint is occupied by other robot, then this robot will consider this waypoint as an obstacle and attempt to find an alternative waypoint. If there is no alternative waypoint can be generated, then the robot should wait for release of the occupied waypoint. In Algorithm 1: For the line 2, focusing on finding the right side of the waypoint for the reason of all vehicles should keep to the right in traffic control problem; for the line 4, the set of e is to deal with the negative influence of sensor error and it should be adjusted in practical applications; the line 4-5 mainly deal with the problem of bridge detection; and the line 8-9 mainly deal with the problem of intersection detection. Each robot has a waypoint list to maintain, whoever has occupied a waypoint will update its waypoint list, and also broadcast to all others. The result by using the proposed algorithm is illustrated in Figure 2.

Our strategy also takes into account the communication

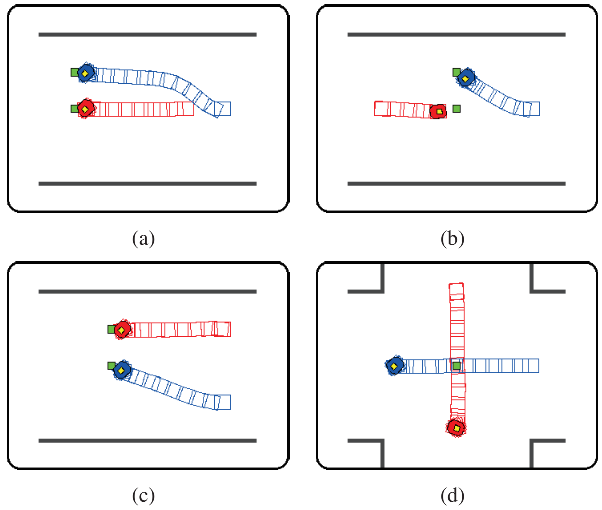

Fig. 2. Robot motion trajectory tracing by using the AWG algorithm. The green small cube represents the waypoint.

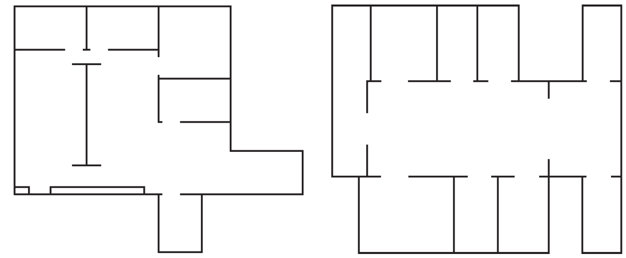

Fig. 3. Two environment maps used in our simulation: map A (left) and map B (right).

bandwidth and the communication range. Since the exchange message for maintaining the waypoint list is very few, and the robots just need to communicate with each other when they ask to pass the same waypoint and therefore should be close enough. In addition, there are two interesting feathers in the proposed algorithm:

- The robots used in MMRS are often miniaturized. The size limitations of these robots also limit the amount of on-board computing power they can carry. Thus, the algorithm applied to the robot should be simple and have a good scalability and realizability. Our algorithm meets this requirements.
- Our algorithm also provides good flexibility. If the working space is large enough, then we can coordinate the motion for more robots (not only two): The generated alternative waypoint is considered as an occupied waypoint for the following robots.

V. Experiments

Our approach has been implemented and evaluated on multi-robot exploration problems where robots should cover an unknown area while building a model of the environment from sensor data, cooperatively. We chose this experimental setup because the exploration problem usually has the context of search and rescue in dangerous environments such as fire and explosion, and the time is a very important consideration in this case [12].

The simulation experiments were conducted in a 2.5D multi-robot simulator called Stage [13], by using a group of virtual Pioneer 2-DX robot equipped with a laser range finder which can provide 361 samples with 180 degrees field of view and a maximum range of 8 meters. Each robot can localize itself based on an abstract localization device which models the implementation of SLAM (Simultaneous Localization And Mapping). The ratio between real-world time and simulation time is about 1:3. All experiments reported in this paper were carried out on a system with an Intel Core 2 Duo E8400 3.00GHz processor, an Intel Q43 Express chipset and two DDR2 800MHz 1024MB dual channel memory.

All the robots maintain a common occupancy grid map with information about the structure of environment, which is used for path planning and obstacle avoidance in real time. We used a different number of robots to conduct several experiments in various environments. Two maps (Figure 3) were used in our simulation which are both structured environment. The exploration team size is varied from 2 to 6 robots. For each team size, 10 experimental runs are performed from different starting positions.

Voronoi-based multi-robot exploration received an increasing attention from the research community in the last decade [14] [15] [16]. We implemented this method for creating the original waypoints in the experiments. In terms of distribution of the original waypoints (task allocation problem), we implemented a trade-based method [17] to dynamic assignment for mobile robots. This method is a variant of the Contract Net Protocol. We evaluate our strategy by using the contrast experiments between the strategy OBO, VFH+, ND, and our waypoint-based method which uses the AWG algorithm described in Section IV. The experimental results are given in Figure 4. We measured the average time of the exploration for each strategy against the size of the robot team. It can be seen that our approach significantly outperforms the approach which does not considrer the waypoint mutual exclusion. Moreover, Figure 4 also shows that, the more the number of robots, the better will the effect of our approach be. Since the probability of the waypoint mutual exclusion increases with the number of robots.

We also counted the average number of the occurrences of waypoint mutual exclusion in each map as shown in Table I. This table shows that the problem of waiting situation is significantly reduced by using our waypoint-based method compared to the others. Since our method is able to plan separate paths for robots by generating the alternative waypoint, unlike the other methods which can only provide a single path for all robots.

VI. Conclusion

In this paper, we presented a decentralized waypoint-based approach for coordinated multi-robot motion planning. The proposed strategy is designed to plan separate kinematic paths

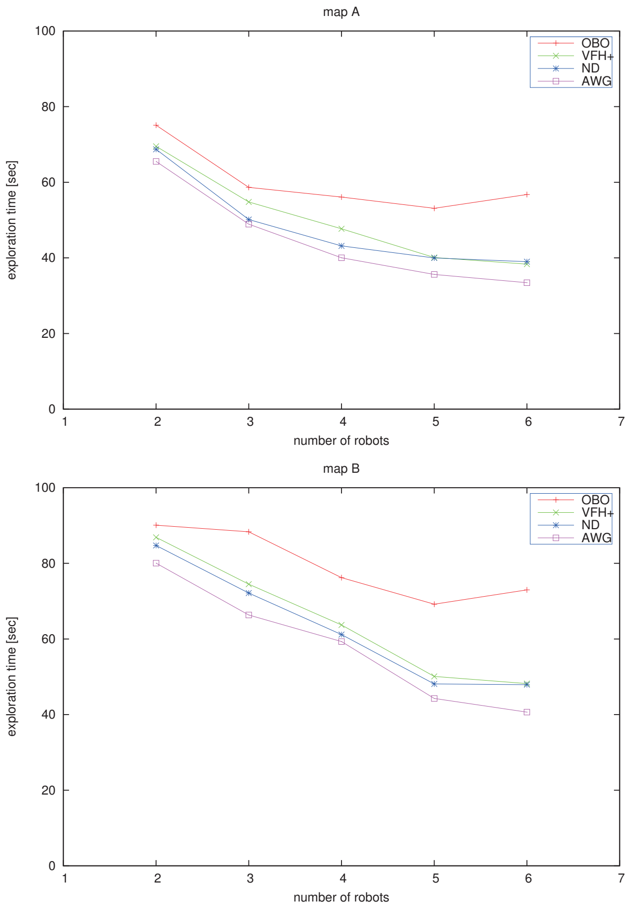

Fig. 4. Exploration time comparison for the map A (top) and the map B (bottom).

TABLE I
STATISTICS OF THE NUMBER OF OCCURRENCES OF THE WAYPOINT MUTUAL EXCLUSION

(a) map A

#robots	2	3	4	5	6
OBO	6.3	7.8	8.4	11.0	13.4
VFH+	6.2	6.7	8.1	10.2	11.6
ND	6.3	5.5	7.8	10.0	10.3
AWG	2.1	2.9	3.4	3.7	3.9

(b) map B

#robots	2	3	4	5	6
OBO	3.2	4.5	5.3	7.0	8.1
VFH+	3.1	3.4	4.1	6.6	7.8
ND	3.1	3.4	4.0	6.6	7.7
AWG	0.0	1.2	2.9	1.9	2.1

to multiple mobile robots to alleviate the waiting situation problems by generating the alternative waypoint (AWG algorithm). We have also discussed briefly the waiting situation problems and illustrated four typical situations when the robots follow: same path in same direction, same path in different direction, different path in same direction, and different path in different direction. The experimental evaluation of the presented approach was conducted on the multi-robot exploration problem and the experimental results demonstrate that, the time needed to accomplish the exploration mission and the occurrences of waypoint mutual exclusion during the exploration have been significantly reduced.

REFERENCES

[1] Z. Yan, N. Jouandeau, and A. Ali Cherif. "Sampling-based multi-robot exploration," in *Proceedings of the Joint 41st International Symposium on Robotics and 6th German Conference on Robotics (ISR/ROBOTIK 2010)*, Munich, Germany, June 2010, pp. 44–49.

[2] P. Švestka and M. H. Overmars, "Coordinated motion planning for multiple car-like robots using probabilistic roadmaps," in *Proceedings of the 1995 IEEE International Conference on Robotics and Automation (ICRA'95)*, Nagoya, Japan, May 1995, pp. 1631–1636.

[3] M. Jäger and B. Nebel, "Decentralized collision avoidance, deadlock detection, and deadlock resolution for multiple mobile robots," in *Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'01)*, Maui, HI, USA, October 2001, pp. 1213–1219.

[4] M. Moors, T. Röhling, and D. Schulz, "A probabilistic approach to coordinated multi-robot indoor surveillance," in *Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'05)*, Alberta, Canada, August 2005, pp. 3447–3452.

[5] L. S. Marcolino and L. Chaimowicz, "Traffic control for a swarm of robots: avoiding group conflicts," in *Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'09)*, St. Louis, MO, USA, October 2009, pp. 1949–1954.

[6] L. S. Marcolino and L. Chaimowicz, "Traffic control for a swarm of robots: avoiding target congestion," in *Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'09)*, St. Louis, MO, USA, October 2009, pp. 1955–1961.

[7] A. Bicchi, A. Fagiolini, and L. Pallottino, "Towards a society of robots," IEEE Robotics and Automation Magazine, vol. 17, no. 4, pp.26–36, December 2010.

[8] V. G. Santos and L. Chaimowicz, "Hierarchical congestion control for robotic swarms," in *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS11)*, San Francisco, CA, USA, September 2011, pp. 4372–4377.

[9] E. W. Dijkstra, "Cooperating sequential processes," in *Programming Languages: NATO Advanced Study Institute, F. Genuys*, Ed. Academic Press, 1968, pp. 43–112.

[10] I. Ulrich and J. Borenstein, "VFH+: Reliable obstacle avoidance for fast mobile robots," in *Proceedings of the 1998 IEEE International Conference on Robotics and Automation (ICRA'98)*, Leuven, Belgium, May 1998, pp. 1572–1577.

[11] J. Minguez and L. Montano, "Nearness diagram (ND) navigation: Collision avoidance in troublesome scenarios," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 1, pp. 45–49, February 2004.

[12] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun, "Collaborative multi-robot exploration," in *Proceedings of the 2000 IEEE International Conference on Robotics and Automation (ICRA'00)*, San Francisco, CA, USA, April 2000, pp. 476–481.

[13] B. P. Gerkey, R. T. Vaughan, and A. Howard, "The player/stage project: Tools for multi-robot and distributed sensor systems," in *Proceedings of the 11th International Conference on Advanced Robotics (ICAR'03)*, Coimbra, Portugal, June 2003, pp. 317–323.

[14] L. Wu, M. A. Garcia, D. Puig, and A. Sole, "Voronoi-based space partitioning for coordinated multi-robot exploration," *Journal of Physical Agents*, vol. 1, no. 1, pp. 37–44, 2007.

[15] A. D. Haumann, K. D. Listmann, and V. Willert, "Discoverage: a new paradigm for multi-robot exploration," in *Proceedings of the 2010 IEEE International Conference on Robotics and Automation (ICRA'10)*, Anchorage, AK, USA, May 2010, pp. 929–934.

[16] A. Cowley and C. J. Taylor, "Rapid multi-robot exploration with topometric maps," in *Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA'11)*, Shanghai, China, May 2011, pp. 1044–1049.

[17] Z. Yan, N. Jouandeau, and A. Ali Cherif. "Multi-robot decentralized exploration using a trade-based approach," in *Proceedings of the 8th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2011)*, Noordwijkerhout, The Netherlands, July 2011, pp. 99–105