Fault-tolerant pattern formation by multiple robots: a learning approach

Jia Wang, Jiannong Cao, Shan Jiang
The Hong Kong Polytechnic University, Kowloon, Hong Kong, China
Email: {csjiawang, csjcao, cssjiang}@comp.polyu.edu.hk

***Abstract*—In the field of multi-robot system, the problem of pattern formation has attracted considerable attention. However, the faulty sensor input of each robot is crucial for such system to act reliably in practice. Existing works focus on assuming certain noise model and reducing the noise impact. In this work, we propose to use a learning-based method to overcome this kind of barrier. By interacting with the environment, each robot learns to adapt its behavior to eliminate the malfunctions in the sensors and the actuators. Moreover, we plan to evaluate the proposed algorithms by deploying it into the multi-robot platform developed in our research lab.**

***Index Terms*—Fault-tolerant; Multi-roobt system; Pattern formation; Reinforcement learning.**

I. Motivation

Recently, it has been a trend in the robotics community to use a group of robots to accomplish the user-defined tasks instead of a single robot [1][2]. In various robotics applications, multiple robots are required to form different shapes in order to enhance coordination efficiency. This problem, called pattern formation problem, is one of the fundamental problems for the multi-robot system. Some examples of such applications are autonomous vehicles in highway [3], area coverage using ground/aerial/underwater robots [4], security and surveillance [5] [6], search and rescue [7], etc.

The natural solutions for multi-robot pattern formation are centralized algorithms, in which the system heavily depends on its central controller. For better scalability, much more research attentions focus on the distributed algorithms that require only local neighbor-to-neighbor information exchange. Roughly speaking, existing distributed methods for pattern formation can be categorized into three classes [8]: consensus based methods [9], leader-follower methods [10], and bio-inspired methods [11]. The consensus algorithms consider all robots to reach an agreement on the centroid of the desired formation. Each robot then specifies a corresponding desired deviation from the center point to achieve formations. However, these methods under-perform in formation maneuvering applications. In leader-follower methods, each group member follows the motion of the group leader to build the formation. The loss of the group leader leads to the entire group to fail. Furthermore, the formation can become disjoint if followers are not able to track the motion of the leader accurately. The bio-inspired methods imitate the physical movement of the magnetic materials employing attractive and repulsive forces to maintain the formation. However, an accurate formation geometry can not be guaranteed. Therefore, above methods are not practical to create a fault-tolerant multi-agent system for pattern formation problem.

In the implementation of practical strategies, the system fault is inevitably due to the dynamic change of environment and noisy sensors input. To address the pattern formation problem concerning the above restrictions, an autonomous fault-tolerant strategy is necessary. Our solution focuses on the design of an autonomous multi-robot cooperative system. The current movements of the other robots in the team are considered as the part of the environment. The robot gains knowledge from the movement and sensor observation, then it plans the optimal action to tolerate system fault. Besides, our proposed method allows the group to continue achieving an objective even in the existence of failure of any group member. Moreover, this approach simplifies the designer's work due to its tolerance regarding sensors and actuators to be finely tuned.

II. Problem Definition

Consider a set of n robots knowing the target shape C and moving freely on an Euclidean plane R^2. All robots are aware of distances d_i and angles a_i to other members through passive sensors but they have no knowledge about the next movements of other robots. Moreover, in practice, these sensor inputs are noisy which leads to a degradation in the pattern formation accuracy. Therefore, our objective is to build a fault-tolerant pattern formation system, which can self-correct malfunctions in sensors and then accurately compose the predefined n-gon shape.

III. Main Challenges

Previous work done by our research group [12] extended distributed convex hull algorithm to form a uniform circle. This method does not require each robot to sense the position of all other robots, thus providing increased efficiency, scalability as well as robustness due to its ability to cooperate with any number of robots. To further strengthen the tolerance of the system, we come up with a learning based approach, which creates new challenges that we need to address.

First of all, the autonomous robots are expected to adapt to any given initial state and start their task. Indeed, this guarantees the fault-tolerance of the system because the disposition after the last fault is regarded as an arbitrary initial state. Since the actions of a robot depend on the motion of other group members, we have to design a proper way that evaluates the action performed by individual robot responding to the extent

978-1-5386-1679-6/17 $31.00 © 2017 IEEE
DOI 10.1109/SRDS.2017.42

of the task achieved by the group. In order to accomplish the development of a system, we divided the research into four phases.

- Conduct the literature review and identify reliable and efficient algorithms for multi-agent reinforcement learning. Due to the system fault increases the dimensionality for both the states and action spaces, conventional learning methods can be slow. Therefore, we should propose a new learning model in which the method coordinates the series-action of the individual agent to achieve group cooperative task.
- Figure out the representation of the state in the multi-agent learning model. This representation should convert the high-dimension sparse spaces from noisy sensor input to low-dimension meaningful space.
- Construct the reward function [13], instead of directly indicating good and bad actions, it should provide useful guidance, which speeds up the learning convergence.

IV. Approaches

We can achieve fault-tolerant pattern formation if each robot sequentially makes optimal decisions. Note that the reinforcement learning shows good performance at solving sequential decision-making problems [14], we attempt to follow this paradigm. We model the environment as a set of observations from individual defective sensor inputs. The numerical feedback called reward is designed based on the similarity between the target shape and its current observation. In particular, multi-agent reinforcement learning typically has two approaches: multiple individual learning and joint action learning. Inspired by the *replay memory* in DQN [15], our approach considers a planning phase before the learning. Considering group information, each robot first generates n-steps experiences under the current policy, then individually updates the new policy with batch of such generated experiences.

A. Group-planning and self-learning approach

At the planing phase, each robot generates experiences in term of the transition triple $(s_t, a_t, r_{t+1}, s_{t+1})$ by the greedy exploration [16], where a_t is the action taken under the state s_t, then the state is transferred to s_{t+1} with r_{t+1} as the reward. The state is the observation of distances and angles to other robots in the team.

At the learning phase, each robot updates the state-action value $Q(s, a)$ with generated experiences as input. Then the action is taken by the greedy policy $a_t = \arg\max Q(s_t, a_i)$. This policy guarantees the future received reward $G = \sum_{i=t}^{T} r_i$ is maximized. The learning iteration ceases after the growth of state-action value is stable.

B. Performance evaluation

The developed algorithms will be evaluated both by simulation and in real applications. For the simulations we will choose Webots[1] as a tool. We will also implement this self-correcting algorithm into the multi-robot platform developed in our research lab.

On the performance evaluation metrics, we will first evaluate the algorithm efficiency, in term of the time taken to learn a suitable policy. Also, we will measure the average received rewards at each iteration, which indicates the way of how to set the learning parameters. In the end, we will analyze the trade-off between system performance and the efficiency cost.

[1] Webots: https://www.cyberbotics.com/overview

V. Acknowledgement

The research is partially supported by the ANR/RGC Joint Research Scheme with number A-PolyU505/12, NSFC/RGC Joint Research Scheme with number N_PolyU519/12 and NSFC with number 61332004.

References

[1] S. Jiang, J. Cao, Y. Liu, J. Chen, and X. Liu, "Programming large-scale multi-robot system with timing constraints," in *Computer Communication and Networks (ICCCN), 2016 25th International Conference on*. IEEE, 2016, pp. 1–9.

[2] S. Jiang, J. Liang, J. Cao, and R. Liu, "An ensemble-level programming model with real-time support for multi-robot systems," in *Pervasive Computing and Communication Workshops (PerCom Workshops), 2016 IEEE International Conference on*. IEEE, 2016, pp. 1–3.

[3] J. Wei, J. M. Dolan, and B. Litkouhi, "Autonomous vehicle social behavior for highway entrance ramp management," in *Intelligent Vehicles Symposium (IV), 2013 IEEE*. IEEE, 2013, pp. 201–207.

[4] A. Howard, M. J. Mataric, and G. S. Sukhatme, "Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem," *Distributed autonomous robotic systems*, vol. 5, pp. 299–308, 2002.

[5] A. Farinelli, L. Iocchi, and D. Nardi, "Distributed on-line dynamic task assignment for multi-robot patrolling," *Autonomous Robots*, pp. 1–25, 2016.

[6] N. Ahmed, J. Cortes, and S. Martinez, "Distributed control and estimation of robotic vehicle networks: Overview of the special issue," *IEEE Control Systems*, vol. 36, no. 2, pp. 36–40, 2016.

[7] A. Macwan, J. Vilela, G. Nejat, and B. Benhabib, "Multi-robot deployment for wilderness search and rescue," *Int. J. Robot. Autom.*, vol. 31, no. 1, 2016.

[8] K.-K. Oh, M.-C. Park, and H.-S. Ahn, "A survey of multi-agent formation control," *Automatica*, vol. 53, pp. 424–440, 2015.

[9] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.

[10] Y. Hong, G. Chen, and L. Bushnell, "Distributed observers design for leader-following control of multi-agent networks," *Automatica*, vol. 44, no. 3, pp. 846–850, 2008.

[11] J. Bai, G. Wen, A. Rahmani, and Y. Yu, "Distributed formation control of fractional-order multi-agent systems with absolute damping and communication delay," *International Journal of Systems Science*, vol. 46, no. 13, pp. 2380–2392, 2015.

[12] S. Jiang, J. Cao, J. Wang, M. Stojmenovic, and J. Bourgeois, "Uniform circle formation by asynchronous robots: A fully-distributed approach," *Manuscript submitted for publication in ICCCN 2017*.

[13] S. Devlin and D. Kudenko, "Plan-based reward shaping for multi-agent reinforcement learning," *The Knowledge Engineering Review*, vol. 31, no. 1, pp. 44–58, 2016.

[14] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. The MIT Press, 1998, vol. 1, no. 1.

[15] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[16] J. Foerster, N. Nardelli, G. Farquhar, P. Torr, P. Kohli, S. Whiteson *et al.*, "Stabilising experience replay for deep multi-agent reinforcement learning," *arXiv preprint arXiv:1702.08887*, 2017.