https://doi.org/10.31256/Sc6Do6C

An End-to-End Task Allocation Framework for Autonomous Mobile Systems

Song Ma
Department of Mechanical Engineering
University College London
London, UK
song.ma.18@ucl.ac.uk

Jingqing Ruan
Institute of Automation
Chinese Academy of Sciences
Beijing, China
ruanjingqing2019@ia.ac.cn

Yali Du
Department of Informatics
King's College London
London, UK
yali.du@kcl.ac.uk

Richard Bucknall
Department of Mechanical Engineering
University College London
London, UK
r.bucknall@ucl.ac.uk

Yuanchang Liu
Department of Mechanical Engineering
University College London
London, UK
yuanchang.liu@ucl.ac.uk

***Abstract*—This work aims to unravel the problem of task allocation and planning for multi-agent systems with a particular interest in promoting adaptability. We proposed a novel end-to-end task allocation framework employing reinforcement learning methods to replace the handcrafted heuristics used in previous works. The proposed framework achieves high adaptability and also explores more competitive results. Learning experiences from the feedback help to reach the advantages. The systematic objectives are adjustable and responsive to the reward design intuitively. The framework is validated in a set of tests with various parameter settings, where adaptability and performance are demonstrated.**

***Index Terms*—task allocation, autonomous system, reinforcement learning**

I. Introduction

Improving the autonomy level of autonomous systems attracts immense interests from both the industry and research communities. The deployments of multi-agent autonomous systems can have the benefits of improved efficiency and increased mission coverage, which is more suitable for addressing complicated tasks. In general, controlling a multi-agent system can be divided into two sub-tasks in sequence: 1) task allocation, which is to assign tasks to different agents and 2) task planning, which is to plan specific task execution sequence for each agent. Such a research problem has been properly discussed in several previous works [1]–[6], where different unsupervised learning algorithms combined with various hand-crafted heuristics were used. However, in most of these studies, task allocation and task planning are carried out in a decoupled way, i.e. no feedback from task planning is provided to guide the task allocation process.

Therefore, in this work, we focus on investigating the task allocation problem by developing a novel coupled framework, where information from task planning stage can be fed back to task allocation in real-time. Specifically, by leveraging the structure of Reinforcement Learning (RL), where the task allocation process is regarded as a RL agent and the task planning process as a RL environment, the proposed framework is successfully equipped with an end-to-end structure with a concise feedback coming from the RL environment to the agent to guide the allocation procedure.

This work was supported by the UCL Dean's Prize scheme and China Scholarship Council.

II. Method

A. Problem Formulation

The overall objective is to plan K independent agent in a collaborative way to visit a set of given waypoints, with each robot starting and ending at the same place, forming closed-loop paths. The paths are formalised as a series of simple cycle graphs, $\{\mathcal{C}_k\}$, where $k = 1, ..., K$ is the number of vehicles deployed in the task. For each simple cycle graph $\mathcal{C}_k$, the graph $\mathcal{C}_k = \left(\mathcal{V}^k, \mathcal{E}^k\right)$, where $\mathcal{V}^k = \{\mathbf{v}_i\}_k, i = 1, .., N_k$ denotes the set of waypoints to be assigned to a particular robot vehicle, and $\mathcal{E}^k = \{l_{ij}\}_k$ denotes a set of lengths of the linked paths between $\mathbf{v}_i$ and $\mathbf{v}_j$. The linked paths topologically forms the edges of the simple cycle graph. Along the cycle graph the elements of V will subsequently have ordered labels from 1 to N_k, where $N_k =\mid \mathcal{V}^k \mid$ is the number of elements in $\mathcal{V}^k$.

In this paper, we use a single-valued metric, i.e. the total travel distance, as the optimisation objective to focus more on the framework design and keep the conciseness. However, further optimisation objectives can be configured according to deployment practices, such as mission completion rate, mission priority and etc. The metric in this work is defined as the summation of the distances of all compartments within $\mathcal{E}_k$.

B. The Reinforcement Learning Agent

In essence, each set of waypoints of the task allocation problem is generally unique. Therefore, the proposed reinforcement learning framework, shown in Fig. 1, should be trained in an

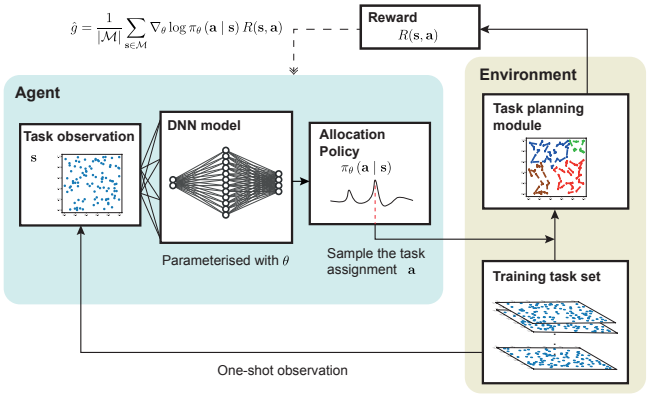

Fig. 1. Diagram of the proposed reinforcement learning framework.

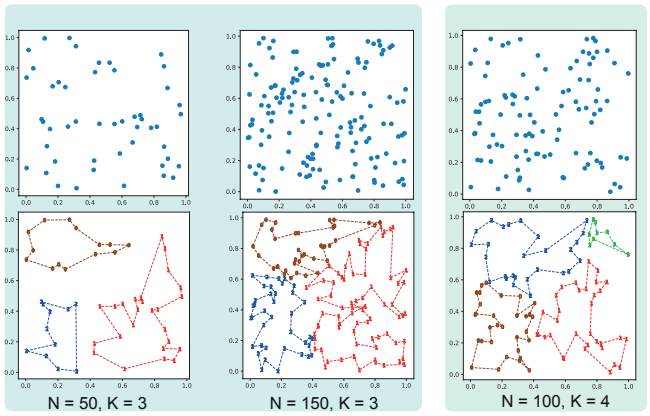

Fig. 2. Three task sets and their corresponding inference results. The left two task sets are inferred by a model with $N = 100$ and $K = 3$ as training parameters, and the right one is inferred by a model with $N = 100$ and $K = 4$.

one-shot manner with unlabelled datasets, which means the episode length of the state-action pairs is $T = 1$. Each task sample consists of N waypoints having a feature dimension of F, which represents the dimension of the workspace. The task samples are fed to the deep neural network (DNN) model parameterised with θ. The DNN then returns the stochastic policy for action sampling, $\pi_\theta(\cdot|\mathbf{s})$, where $\mathbf{s}$ denotes the observed task sample. The allocation action is sampled from the stochastic categorical distribution as (1).

$$\mathbf{a} \sim \pi_\theta(\cdot|\mathbf{s}) \tag{1}$$

Then the task planning module determines a detailed plan based on the allocation action. The detailed plan guides the agents to visit the assigned task waypoints in a specific order. The outcomes are formed into closed-loop paths for the agents. The reward of the reinforcement learning framework is set to be the negative value of the total distance of the planned paths, which is fed back to the agent for model optimisation following policy gradient paradigm shown in (2), where $\mathcal{M}$ is a batch of training data, and $R(\mathbf{s}, \mathbf{a})$ is the reward.

$$\hat{g} = \frac{1}{|\mathcal{M}|} \sum_{\mathbf{s} \in \mathcal{M}} \nabla_\theta \log \pi_\theta(\mathbf{a} \mid \mathbf{s}) R(\mathbf{s}, \mathbf{a}) \tag{2}$$

III. Experiments and Results

This section presents a set of experiments to demonstrate the performance of the proposed framework with randomly generated task sets. The task set represents a set of task samples $\mathbf{s}$, each of which contains N waypoints.

The work space is normalised as a 1×1 square, which consists of 50, 100, 150 generated waypoints. The number of agents deployed for the simulated mission are set to be 3 and 4. The reinforcement learning framework features the adaptability to the number of tasks waypoints N, which can be regarded as a hyper-parameter. Due to the adaptability of the designed framework, training only need to consider two datasets: (1) 100-waypoint task assigned to 3 robots, (2) 100-waypoint task assigned to 4 robots, with inference to be carried out for other cases with different waypoints. Both training sets contain 1,000,000 task samples and are loaded with the batch size of 32. Both RL loss and reward valus converged.

For the inference cases with 3 robots to allocate, two different scenarios with $N = 50$ and $N = 150$ are tested. For the case with 4 robots, the test scenario remains identical to the training set, i.e. $N = 100$. Based upon allocation results, the task planning can output paths representing sequences to visit each waypoint as shown in Fig. 2.

IV. Conclusion

We proposed an end-to-end reinforcement learning framework for multi-agent autonomous systems. The novel structure coupled the task allocation and task planning stages within the framework using a feedback mechanism. This feedback can agilely adapt to the optimisation goals in different scenarios. We also presented several simulation results revealing the adaptability of the proposed framework.

References

[1] S. Ma, W. Guo, R. Song, and Y. Liu, "Unsupervised learning based coordinated multi-task allocation for unmanned surface vehicles," *Neurocomputing*, vol. 420, pp. 227–245, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231220314399

[2] Y. Liu, R. Song, R. Bucknall, and X. Zhang, "Intelligent multi-task allocation and planning for multiple unmanned surface vehicles (usvs) using self-organising maps and fast marching method," *Information Sciences*, vol. 496, pp. 180–197, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020025519304323

[3] Y. Liu, E. Anderlini, S. Wang, S. Ma, and Z. Ding, "Ocean explorations using autonomy: Technologies, strategies and applications," in *Offshore Robotics*, S.-F. Su and N. Wang, Eds. Singapore: Springer Singapore, 2022, pp. 35–58.

[4] C. Cunningham and R. Roberts, "An adaptive path planning algorithm for cooperating unmanned air vehicles," in *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*, vol. 4, 2001, pp. 3981–3986 vol.4.

[5] A. Khamis, A. Hussein, and A. Elmogy, "Multi-robot task allocation: A review of the state-of-the-art," *Cooperative robots and sensor networks 2015*, pp. 31–51, 2015.

[6] G. A. Korsah, A. Stentz, and M. B. Dias, "A comprehensive taxonomy for multi-robot task allocation," *The International Journal of Robotics Research*, vol. 32, no. 12, pp. 1495–1512, 2013. [Online]. Available: https://doi.org/10.1177/0278364913496484