DRL-Based Task Management for Robots in Edge–Cloud Infrastructures

Mahyar Karami, Farimehr Zohari, and Masoud Ardakani
Electrical and Computer Engineering Department, University of Alberta, Edmonton, Canada.
Email: mahyar2@ualberta.ca

INTRODUCTION

Robots increasingly rely on computation-heavy tasks such as perception, simultaneous localization and mapping (SLAM), and motion planning. To address the limitations of onboard processors, edge–cloud computing enables robots to offload workloads to nearby edge servers or powerful remote cloud servers, thereby reducing computation delay and improving real-time performance. However, when multiple robots operate simultaneously, shared wireless bandwidth, time-varying channel conditions, and dynamic task arrivals make efficient task management a non-trivial problem [1]. In this paper, we propose a **deep reinforcement learning (DRL)** framework for joint computation offloading and bandwidth allocation in multi-robot systems. Each robot dynamically decides what portion of its tasks should be executed locally, offloaded to the edge, or forwarded to the cloud, while the available communication bandwidth is split between uplink and downlink transmissions. By modelling the system as a Markov decision process (MDP), the proposed DRL-based approach adaptively minimizes task response time under dynamic and uncertain conditions, offering an intelligent solution for real-time robotic task management in edge–cloud environments.

MATERIALS AND METHODS

We consider a three-tier robot–edge–cloud system. A set of N robots generate computational tasks such as perception and control at rates θ_i , $i \in \{1, \dots, N\}$. Each robot has limited onboard computing capacity μ_i , while an edge server, connected through an access point (AP), provides higher but finite processing capacity μ_E. The edge can also forward tasks to a remote cloud server with abundant resources μ_C. Robots may process tasks locally or offload them. The offloading ratio of robot $a_i \in [0,1]$, where $(1 - a_i)\theta_i$ tasks are executed locally and $a_i\theta_i$ are offloaded. The edge forwards a fraction a_E of the aggregated received load $\lambda = \sum_{i=1}^{N} a_i\theta_i$ to the cloud. Task input data (D_u) is transmitted via the uplink channel of rate u, while results (D_d) will be returned through the downlink channel of rate d. The total response time includes: (i) local execution delay, (ii) uplink transmission delay, (iii) edge execution delay, (iv) cloud execution, and (v) downlink transmission delay [2]. The optimization problem aims to minimize the average response time while maintaining stability of local queues, communication links, and the edge server; the full formulation is given in [2]. We formulate the joint task offloading and bandwidth allocation problem as an MDP. At each decision epoch, the **state** captures system dynamics including task arrival rates of robots, their computing capacities, the current load of the edge server, the uplink and downlink channel conditions, and past allocation decisions. The **action** specifies the offloading ratios of robots, the forwarding ratio at the edge, and the bandwidth split between uplink and downlink channels. The **reward** is defined as the negative of the average response time, encouraging the agent to minimize overall delay while respecting resource limits.

RESULTS AND DISCUSSION

The proposed model enables analysis of key trade-offs in robot–edge–cloud task management. When robots process tasks locally, response time grows quickly with task arrival rates due to limited onboard processing capacity. In contrast, full offloading alleviates the local load but introduces uplink congestion and longer transmission delays. The formulation highlights the importance of jointly optimizing offloading ratios and bandwidth allocation: allocating more uplink resources benefits high-data tasks (e.g., perception), while increasing downlink capacity favours tasks with larger result sizes. A DRL-based policy effectively learns these dynamics adaptively, consistently outperforming static baselines such as local-only or full offloading.

CONCLUSIONS

We proposed a robot–edge–cloud system model and formulated offloading and bandwidth allocation as an MDP solved with DRL. The approach enables adaptive task management under dynamic workloads and network conditions, consistently achieving lower response times and better resource utilization compared to static baselines.

REFERENCES

[1] M. Andronie, et al., "Big data management algorithms, deep learning-based object detection technologies, and geospatial simulation and sensor fusion tools in the internet of robotic things," ISPRS Int. J. Geo-Inf., vol. 12, no. 2, p. 35, 2023, doi: 10.3390/ijgi12020035.

[2] K. Guo, M. Yang, Y. Zhang and J. Cao, "Joint Computation Offloading and Bandwidth Assignment in Cloud-Assisted Edge Computing," in IEEE Transactions on Cloud Computing, vol. 10, no. 1, pp. 451-460, 1 Jan.-March 2022, doi: 10.1109/TCC.2019.2950395.