Edge-Cloud Cooperative Intelligence for Computation Offloading in Connected Living

Chaogang Tang, *Member, IEEE*, Huaming Wu, *Senior Member, IEEE*, and Ruidong Li, *Senior Member, IEEE*

Abstract

In the context of the interconnected living environment, an extensive volume of data is continually generated by Internet of Things (IoT) devices, which requires proficient storage, real-time processing, and analysis. Efficient data processing and management play a pivotal role in healthcare for monitoring and detecting abnormal health conditions, especially in the era of the COVID-19 pandemic. Computing paradigms such as cloud intelligence and edge intelligence are considered to be efficient in achieving such goals. Owing to their own shortcomings (e.g., long latency for cloud intelligence and insufficient computing capability for edge intelligence), we propose an efficient edge-cloud cooperative intelligence (EC2I) for data processing and task offloading. Particularly, we try to harness the computing capability of high-end IoT devices and treat them as edge devices. We aim to address the central question of how and where computational tasks are executed within the realm of connected living. Furthermore, we delve into several opportunities and challenges, thereby shedding light on potential avenues for future research in this domain.

I. Introduction

The recent proliferation of Internet of Things (IoT) has tremendously accelerated the reformation of our daily lives. This advancement has facilitated increased interactions between individuals and their surrounding environments, thereby establishing connected living as the prevailing norm [1]. In this constantly interconnected world, everybody can utilize an array of connected devices, e.g., wearable watches, smart bracelets, smartphones, and medical sensors, to fulfill a multitude of purposes. These purposes encompass various aspects such as entertainment, health management, social activities, and more. In the context of the global spread of COVID-19 in recent years, connected living has played a crucial role in terms of health monitoring and the detection of vital bodily signals [2]. A substantial volume of data emanates from diverse data sources and is continuously collected. Managing this data necessitates storage, real-time processing, and analysis, which holds utmost significance in healthcare for the monitoring and detection of abnormal health conditions [3], [4].

Computational intelligence technologies, relying on biologically motivated computing paradigms, have achieved remarkable milestones in various domains. For instance, artificial intelligence (AI) algorithms, deployed in cloud-based physical infrastructure for addressing insufficient computing capacity issues in model training and feature mining, have made enviable breakthroughs in domains ranging from image processing to natural language processing. The combination of AI technologies and the cloud, also termed the *cloud intelligence*, can flexibly integrate the storage, computing, learning and reasoning capabilities of both AI and cloud computing. However, the drawbacks of cloud intelligence, such as long latency, high bandwidth, and potential privacy exposure [5], may limit its widespread applicability in latency-sensitive connected living environments.

Edge computing, which extends across distributed infrastructures, has emerged as an indispensable component within the broader IoT landscape, encompassing applications in smart workspace, smart homes, and smart cities. It is not surprising to bring intelligence to edge computing, forming a novel paradigm, referred to as the *edge intelligence* [6]. It can overcome the drawbacks of cloud intelligence to a certain extent. However, edge computing is not rich enough in computing and storage resources, which limits the system performance of edge intelligence. What's worse, the edge computing paradigm remains incomplete in two critical aspects. Firstly, it does not fully harness the computing capabilities inherent in IoT devices. Secondly, it seldom takes into account the diversity of edge devices, which can result in substantial performance variations. Given the heterogeneous nature of the computing continuum, several challenges persist, including how to leverage the computing potential of end devices, where to offload computational tasks, and how to facilitate effective cooperation between the edge and the cloud.

In view of the enormous potential to provide diverse computing resources for connected living by combining the merits of edge intelligence and cloud intelligence, we make our efforts to extend and strengthen this intelligence-enabled continuum by constructing *edge-cloud cooperative intelligence* (EC2I). EC2I with a well-defined collaboration between edge servers and cloud centers [7], [8], is expected to enhance the performance of edge intelligence, by deploying and executing AI algorithms such as deep neural networks (DNN) models in a

This work was supported in part by the National Natural Science Foundation of China under Grant 62071327 and in part by Tianjin Science and Technology Planning Project under Grant 22ZYYYJC00020.

Digital Object Identifier: 10.1109/MIOT.2025.3581933
Date of Current Version: 30 September 2025
Date of Publication: 23 June 2025

Chaogang Tang is with the School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221116, China; Huaming Wu (corresponding author) is with the Center for Applied Mathematics, Tianjin University, Tianjin 300072, China; Ruidong Li is with the Institute of Science and Engineering, Kanazawa University, Kanazawa 920-1192, Japan.

separated-layer fashion. It aims to meet a broad spectrum of requirements associated with various IoT tasks and applications. Additionally, we strive to address the interrogation of how and where the computation is undertaken within the connected living environment, while meeting diverse performance criteria and specific requirements. In particular, the major contributions can be summarized as follows:

- Considering the differences in computing, caching, and networking capabilities among edge devices, we classify these devices into three distinct categories: Mini-edge, Micro-edge, and Macro-edge. This categorization aims to accommodate and leverage the diverse and flexible nature of IoT devices.
- We offer an architectural overview of a multi-granularity EC2I architecture, which incorporates three types of offloading operations. These operations are introduced with the purpose of fully leveraging the computing capabilities of end devices.
- We present a deep reinforcement learning (DRL) based framework for determining where to undertake the computation for connected living. The average response latency for all the tasks is adopted as the metric for performance evaluation.
- We highlight the opportunities and challenges within the computing continuum for connected living, with the aim of providing insights into potential directions for future research in this field.

II. Multi-Granularity Edge-to-Cloud Continuum Architecture

We propose an EC2I architecture that comprises three distinct layers, namely, the IoT layer, the edge layer, and the cloud layer, as illustrated in Fig. 1.

- IoT layer: The IoT layer comprises a large spectrum of sensors dedicated to signal detection and information gathering. These sensors include, for example, wearable devices and embedded medical sensors within the connected living environment. Some of these sensors, such as smartwatches and bracelets, are equipped with computing capabilities for data preprocessing and basic task execution.
- Edge layer: The edge layer encompasses a variety of edge devices, ranging from smart IoT devices with high-performance processors and communication modules to vehicles equipped with cognitive and intelligent capabilities. Note that edge intelligence enables edge devices with relatively powerful computing capabilities in this layer to run un-complex lightweight artificial neural networks for data processing, task execution and decision making. Such devices include but are not limited to smart cell phones, vehicles, and UAVs.
- Cloud layer: The cloud layer integrates and consolidates various high-performance computing resources located in remote data centers. These resources are delivered on demand through the use of virtualization technologies. In addition, the cloud center can take more roles in cloud intelligence, e.g., training complex DNN models.

A. Edge-Cloud Cooperative Intelligence

Some IoT devices concentrate on the acquisition of execution results, especially in application scenarios like virtual reality and augmented reality. In these cases, the data collected is outsourced to the computing continuum in the form of IoT tasks.

The integration of AI and caching technologies can enhance the overall performance of task execution in such scenarios. However, it's worth noting that the extent of improvement is not always readily apparent. The reason for this variability lies in the nature of IoT tasks, which often rely on specific contextual factors. These factors encompass context-related task inputs, user-specific parameters, and distinct requirements. Consequently, the characteristics of IoT tasks make it challenging to generalize features derived from big data through AI technologies. Furthermore, cached execution results often remain underutilized due to the context-dependent nature of these tasks.

Meanwhile, certain IoT devices prioritize data analysis over receiving processing results, as their primary objective is not feedback but proactive response. In such cases, the processed and analyzed data is mainly used to trigger accident prevention mechanisms. For instance, wearable or built-in medical sensors for ECG signal detection send the gathered medical data to the edge for early heart disease diagnosis. If the data is abnormal via detection, the corresponding prevention measures can be activated promptly to ensure people's health. In this vein, edge intelligence will play an indispensable role. By embedding AI functions into the edge and even within IoT devices themselves, rapid decision-making regarding various analysis-oriented IoT data becomes feasible. This capability enables the system to respond promptly to critical situations, enhancing safety and well-being in connected living environments.

AI-based approaches, spanning across distributed infrastructures, typically encompass two key components: cloud-based model deployment and training, and edge-based inference. This collaborative paradigm within the computing continuum is effectively depicted in Fig. 2, where the remote cloud center is responsible for crucial tasks such as model development, deployment, and training. In contrast, the edge devices utilize these pre-trained network models for data inference and analysis.

B. Three Distinct Categories for Edge Devices

Edge intelligence enhances the performance of edge computing systems by deploying AI algorithms on edge devices to enable rapid data processing and real-time feedback. The feasibility of this vision largely depends on the computing capabilities of the edge devices. In this article, we focus primarily on the provisioning of computing resources at the edge. It is well recognized that IoT devices, when equipped with sufficient computing and communication capabilities, can effectively serve as edge devices and contribute to the overall system performance.

However, owing to the diversity among them and the variations in their performance when provisioning computing resources, we still need to classify these edge devices to fully accommodate and leverage the diverse and flexible nature of IoT devices. Particularly, the edge devices in this article are classified into three distinct categories, i.e., Mini-edge, Micro-edge and Macro-edge, respectively.

1) Mini-Edge: The concept of Mini-edge centers around individuals within the connected living environment and primarily encompasses high-end IoT devices equipped with multi-core processors. Conventionally, these IoT devices have not been

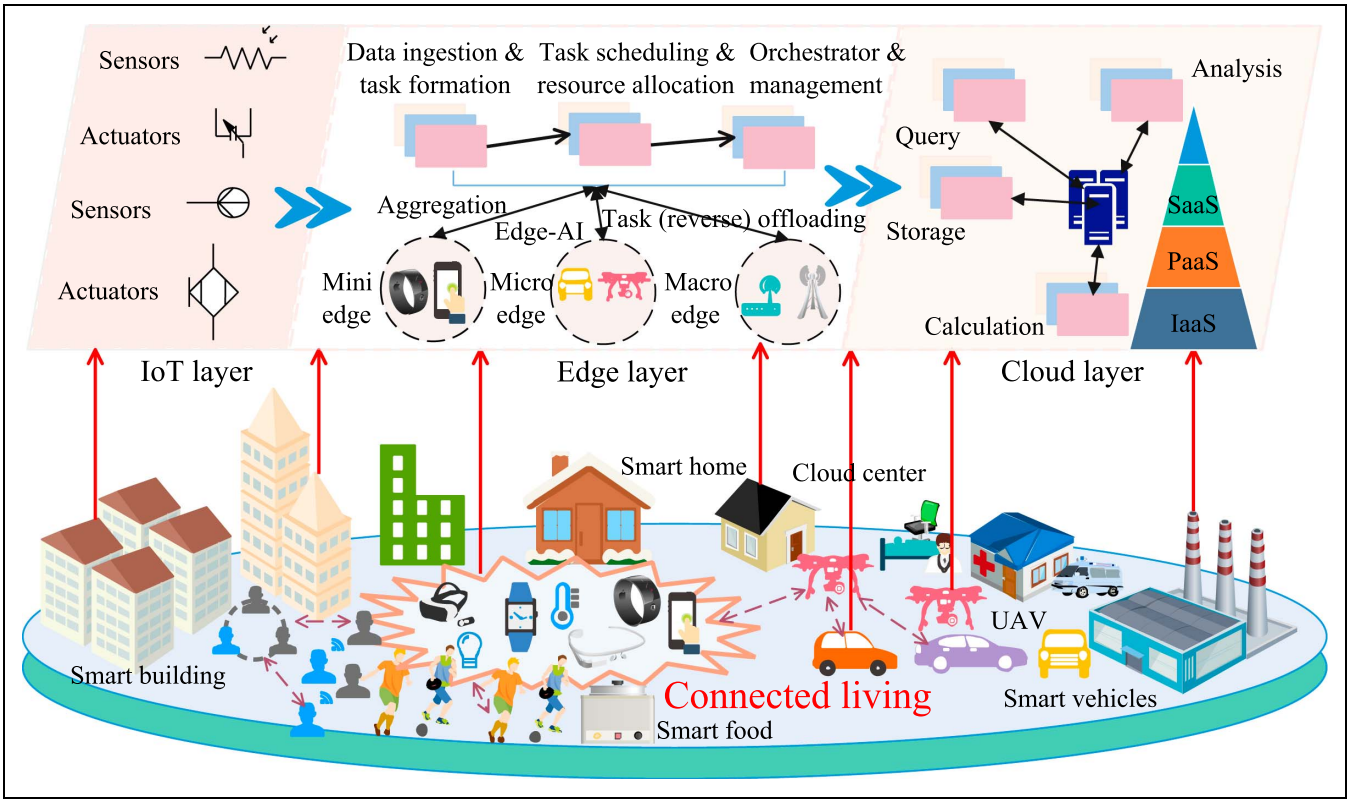

FIG. 1. An architecture of edge-to-cloud continuum for connected living.

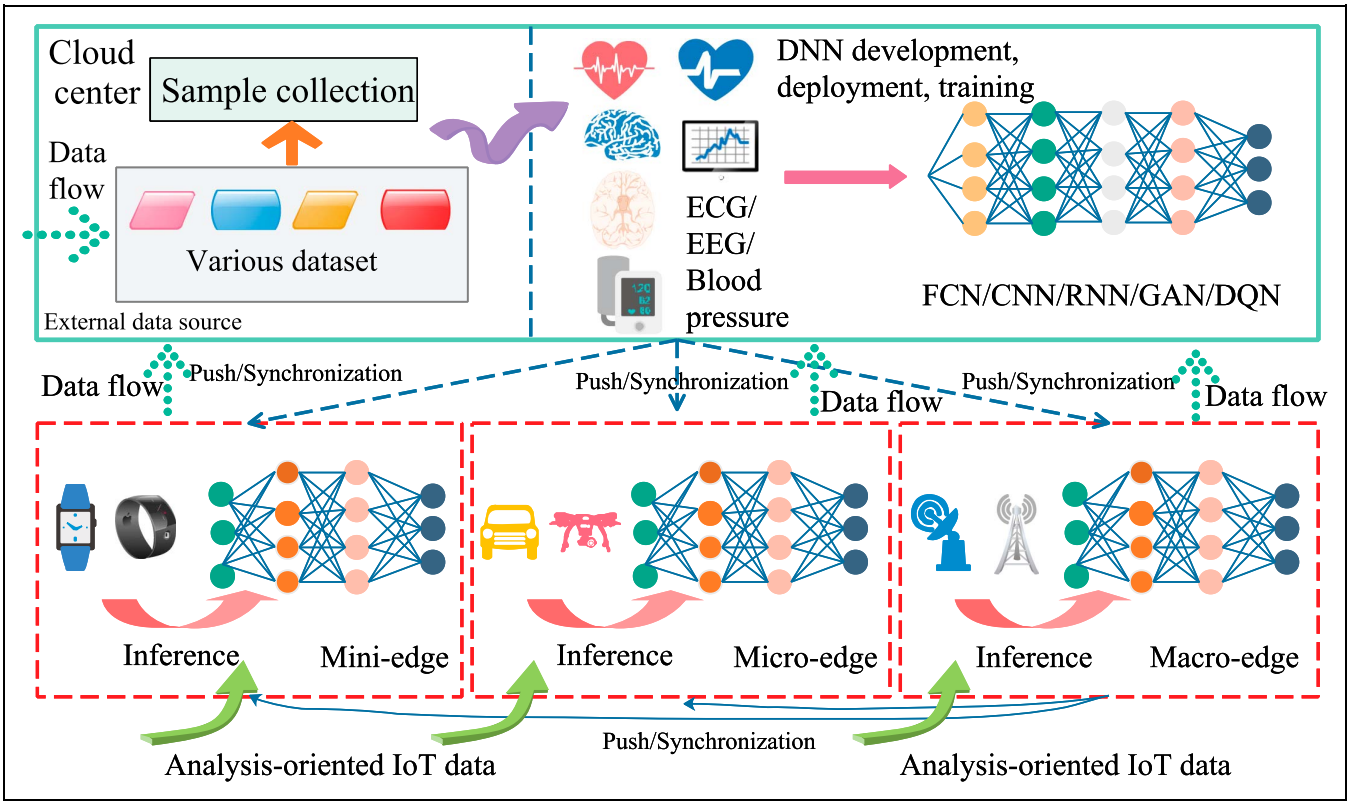

FIG. 2. Collaboration in EC2I development and edge-based inference in connected living.

categorized within the traditional edge layer of the existing computing continuum. Instead, they have often been viewed as data sources or task generators. However, the rapid evolution of IoT technology has empowered microprocessor-integrated IoT devices to become more intelligent and powerful entities. They can now not only meet increasingly complex functional demands but also emerge as potential

providers of computing resources. As a matter of fact, the computing resources within these devices are frequently underutilized. Thus, we propose to fully exploit these over-provisioned resources to enhance the efficiency of the computing continuum. Specifically, we advocate for the complete utilization of these over-provisioned resources by enabling the execution of tasks from edge devices on these resource-idle IoT devices. This practice is referred to as "task reverse offloading" within the context of this article. Note that edge devices belonging to this category can also deploy well-trained, lightweight AI DNN models for data processing and fast event responses.

2) Micro-Edge: Micro-edge mainly refers to edge devices that possess more abundant computing resources compared to Mini-edge. Examples of such Micro-edge devices include smart vehicles and uncrewed aerial vehicles (UAVs). The inclusion of powerful On-Board Units (OBUs) equips vehicles with the capability for environment sensing, intelligent decision-making, task execution, and data analysis. This phenomenon has given rise to concepts such as vehicular fog computing (VFC) and vehicular edge computing (VEC). Similarly, the deployment of onboard computers on UAVs has led to the emergence of aerial edge computing (AEC). Both "edge on the wheels" and "edge in the sky" are worthy of further exploration for connected living. For instance, in connected communities, a significant portion of computing resources for parked vehicles often remains idle. Leveraging these idle resources collectively as a resource pool for edge computing presents substantial opportunities for enhancing the capabilities and services offered in the connected living ecosystem.

3) Macro-Edge: Macro-edge refers to the traditional edge server infrastructure that brings cloud resources sink into the network edge, such as accessible computational access point (CAP) and base station (BS) with strong computing power, large storage, and rich networking resources. The edge servers can share enormous backhaul pressure and lower response latency, by undertaking the computation that is originally offloaded to the cloud center. Macro-edge has the richest computing, storage and networking resources, in comparison with Mini-edge or Micro-edge. It is foreseeable that, though with slow progress, edge computing will become the workhorse for connected living in smart cities.

III. Computation Offloading for Connected Living

Current research efforts often tend to focus on either optimizing resource allocation from the perspective of the edge server or enhancing response latency from the standpoint of IoT devices. On no account can we take computation offloading and task execution outside the IoT devices for granted. The driving motivation behind computation offloading stems from the energy constraints and limited computational capacity inherent in IoT devices [9], [10], [11], [12].

Within the proposed architecture, there are at least five potential candidates where generated tasks can be offloaded and executed. These candidates include the IoT devices themselves, as well as Mini-edge, Micro-edge, Macro-edge, and the cloud center. Meanwhile, there exist several ways to offload tasks, such as partial offloading and binary offloading. Importantly, these issues are often intricately interconnected, contributing significantly to the complexity of the decision-making process and resource allocation when it comes to task offloading.

For all intents and purposes, addressing the computation offloading should basically understand two fundamental issues, i.e., how and where to undertake computation for connected living within the proposed edge-to-cloud continuum. As a consequence, we try to answer two questions revolving around computation offloading in connected living.

A. How to Undertake Computation

There are different ways to undertake computation from different angles. In this article, we briefly discuss three kinds of task offloading in the following.

- Binary offloading: The IoT devices are merely treated as task generators, irrespective of their computing capabilities. Once tasks are generated, they are completely offloaded to the computing continuum for execution. This outsourcing operation effectively transfers the task workload from the local host to the edge or cloud infrastructure. Subsequently, IoT devices await the resulting feedback from the computing continuum. In the context of the connected living environment, numerous sensors and IoT devices necessitate binary offloading for data processing. For instance, medical sensors embedded under the skin play a crucial role in detecting cardiovascular and cerebrovascular diseases. These sensors generate data that requires processing by external computing resources due to the complexity of the tasks involved.
- Partial offloading: Partial offloading is a strategy that considers the computing capabilities of IoT devices. In this approach, IoT devices are capable of performing specific computations without compromising their primary functions, such as environmental sensing, data gathering/fusion, and task generation. Consequently, the workload of the task is distributed between the local IoT device and the computing continuum. Furthermore, the results of the computations from both sides need to be combined to form a complete feedback result. Achieving this typically involves data transmission and result integration, tasks that can be efficiently handled by the edge or cloud due to their robust computing capabilities. As a result, careful consideration must be given to partitioning the computation, taking into account two different execution times as well as the time required for data transmission.
- Reverse offloading: The driving force behind the concept of task reverse offloading lies in the observation that the computing resources of IoT devices are often over-provisioned, allowing them to offer computing services when these resources are not actively engaged. In the typical process, IoT devices are primarily responsible for sensing and gathering data. Once the data is collected, it is transmitted to the edge. Subsequently, data fusion and task generation occur at the edge. This approach differs from previous cases where IoT tasks were generated directly at the IoT device. This shift in task generation is both reasonable and justifiable, particularly when data fusion and task generation involve complexity and stringent time

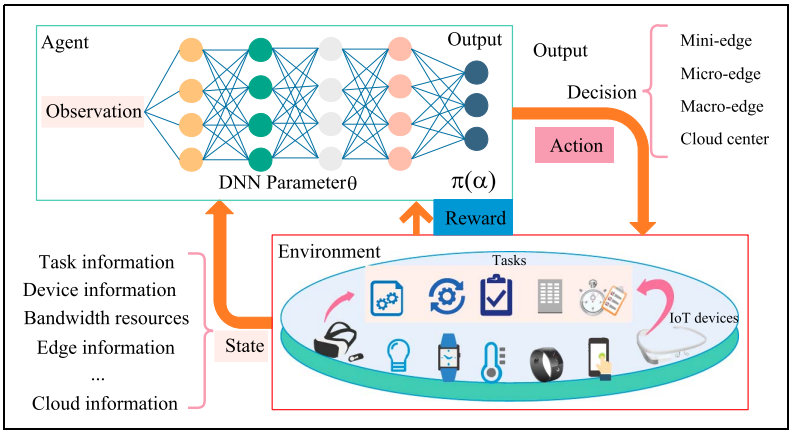

FIG. 3. A DRL-Based framework for determining where to undertake the computation for connected living.

constraints. When tasks are generated at the edge, they can be transmitted back to the IoT devices for computation, a procedure referred to as task reverse offloading. It's noteworthy that both binary offloading (where the entire task is offloaded) and partial offloading (where only a portion of the task is offloaded) can be applied during the task reverse offloading process. Furthermore, within the scope of this article, IoT devices belonging to the Mini-edge category are identified as suitable candidates for task reverse offloading.

B. Where to Undertake Computation

The decision regarding where to offload IoT tasks within the edge-to-cloud continuum is inherently challenging. This challenge arises from several factors, including the diverse performance and specific requirements associated with IoT tasks, as well as the significant heterogeneity among edge devices in terms of computing, storage, and networking capabilities. Even when the method of task offloading is predetermined, determining the optimal location for offloading tasks remains a complex and dynamic process.

The fusion of AI with edge computing gives rise to the concept of edge artificial intelligence (edge AI). The primary objective of edge AI is to enhance rapid decision-making and facilitate various resource allocations within the computing continuum. For instance, DRL has attracted substantial attention across a wide range of applications, including various optimization problems, ranging from resource allocation to task offloading [13], [14], [15]. In DRL, an agent is trained to acquire knowledge and experience by interacting with its environment. Subsequently, the agent adopts the currently "best" action to explore the environment and obtain a reward.

Specifically, we present a DRL-based framework for determining where to undertake the computation for connected living, as depicted in Fig. 3. The essential components of DRL, namely state, action, and reward, are elaborated upon as follows:

State: The state in DRL is a set of potential conditions that characterize the connected living environment. In this article, the state can be effectively represented as a multi-dimensional vector consisting of task information, device specifications, bandwidth resources, and available edge/cloud information.

Action: The definition of the action space depends upon the decision variables associated with the specific optimization problem at hand. Generally, the agent strives to learn a mapping function or a set of neural network parameters that can effectively map the state space to the action space, with the aim of maximizing the expected cumulative reward. For instance, when dealing with partial offloading, the decision variables on where to undertake the computation and the decision variables for resource allocation are usually tightly coupled. Thus, the action space can be composed of two distinct components. One component comprises discrete values that specify where the computation should occur, while the other component consists of continuous values that determine the quantity of computational resources allocated to the IoT task.

Reward: Upon observing the current state, the agent selects an appropriate action from the defined action space. Then, the agent receives a reward from the environment. The reward function usually depends on the specific objective function associated with the optimization problem. For instance, if the objective function is to maximize the system utility within the computing continuum, the function formulated for system utility maximization can be regarded as the reward function.

Practical implementation of the DRL-based strategy requires seamless coordination and collaboration among the various entities within the computing continuum. Owing to wide coverage and powerful computing capabilities, a Base Station (BS) located at the Macro-edge can be strategically chosen to host an agent tasked with interacting with the environment. This agent's responsibilities encompass action design and the subsequent dissemination of these actions among IoT devices, encompassing the Mini-edge and Micro-edge layers. The process of deploying and executing this DRL-based strategy can be outlined as follows:

- First, the environment's state is generated through the aggregation of various information.
- Then, given the current state, the agent employs deep neural networks (DNN), e.g., DQN or DDPG, to determine the next action. The specific policy update mechanism depends on the adopted network model. For instance, in the case of DDPG, the target policy value and target Q value updates are overseen by the Actor-T and Critic-T components within the target network. The action encompasses both discrete values, which represent decisions regarding offloading destinations, and continuous values, which pertain to resource allocation schemes for IoT tasks.
- Third, the action is propagated to edge devices at the edge layer, where they prepare for resource allocation based on the action, and to the IoT devices with offloading requests, which make preparations for task offloading in accordance with the action. After task offloading and resource allocation, information from the involved entities is collected to create a new state. In response to this state, the environment provides immediate feedback in the form of a reward to the agent. The state-based policy can then be updated based on this state. For instance, if the current reward surpasses the previously recorded reward, signifying it as the best, the recorded reward is updated to match this new value. The convergence of cumulative rewards signifies the success of the training process for offloading decision-making and resource allocation.

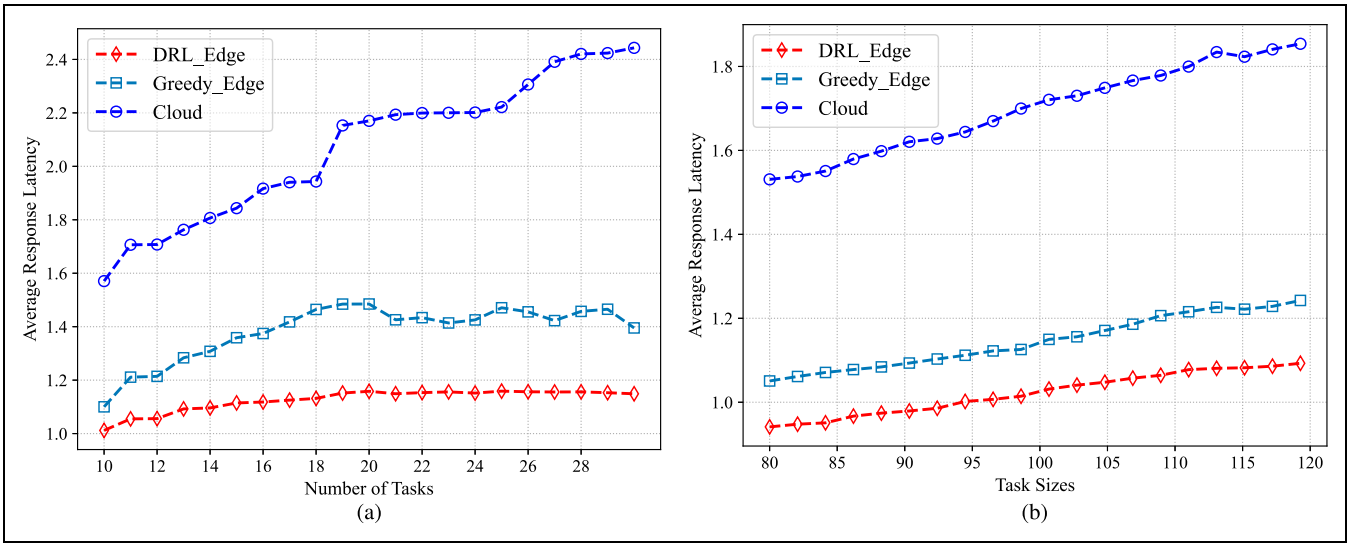

FIG. 4. Evaluation on average response latency for different approaches. (a) Latency comparison with different number of tasks (b) latency comparison with different task sizes.

IV. Performance Evaluation

In this study, our primary objective is to investigate the intricacies of determining when and where computation takes place within the context of connected living, and we achieve this through comprehensive simulations. An edge computing-enabled connected living environment is simulated, which comprises deadline-specified computation tasks generated by various IoT applications in connected living. The environment incorporates three edge devices and a cloud center. It is noted that the performance-varied edge devices belong to the three aforementioned categories, respectively. The cloud center can provide rich computing resources. To simplify the simulation, we adopt a binary offloading strategy in this simulation, and the tasks can be executed by the edge devices or the cloud center.

Three approaches have been employed and are subject to a comparative analysis based on their average response latency for all the offloaded tasks. These approaches include the DRL-based approach, the greedy approach, and the cloud-based approach, respectively. The greedy approach consistently selects the edge device with the most abundant computing resources as the designated offloading destination. The cloud-based approach systematically offloads tasks to the cloud center. For the purpose of these simulations, the number of tasks and their respective sizes are randomly generated within predefined ranges, specifically [10, 30] for the number of tasks and [80, 120] for task sizes. The simulation results are depicted in Fig. 4, where Fig. 4(a) and 4(b) illustrate the average response latency for all the tasks generated during the optimization period, when the number of tasks and the task size vary, respectively.

In terms of average response latency, the DRL-based approach outperforms the other two approaches, while the cloud-based approach exhibits the highest latency among the three. Furthermore, as illustrated in Fig. 4(a), it becomes evident that an increase in the number of tasks leads to a corresponding rise in average response latency, regardless of the approach employed. This finding underscores the challenges posed by a substantial volume of tasks on the computing capabilities of edge devices. The increase in task volume leads to higher queueing time, which in turn contribute to the overall increase in average response latency. Moreover, the size of IoT tasks plays a significant role in influencing average response latency, as demonstrated in Fig. 4(b). Across all three approaches, it is evident that as task sizes increase, the average response latency experiences a concurrent increase. This finding underscores the sensitivity of response times to the scale of computational tasks, reinforcing the critical importance of efficient task allocation and resource management within the connected living environment.

V. Challenges

The edge-to-cloud continuum stands as a pivotal enabler for various applications within the realm of connected living. However, it is essential to acknowledge that certain challenges persist, demanding prompt attention and innovative solutions. In this context, we delve into a discussion of some of these challenges, with the intent of providing guidance for potential future directions within this dynamic field.

A. Security and Privacy Issues in Connected Living

Among all the issues that must be addressed in connected living, security and privacy may be the most urgent ones. On the one hand, IoT tasks frequently involve personal information that can be easily inferred and predicted with the aid of a robust computing continuum. In scenarios where edge devices, particularly those within the Mini-edge category, exhibit malicious behavior, the repercussions can range from privacy breaches during service provisioning in the best-case scenario to tampering with execution results and causing substantial harm to the connected living environment in the worst case. On the other hand, when it comes to IoT data analysis, malicious attackers have the potential to manipulate analysis results, posing a significant threat to people's well-being in the context of connected living. Malicious attackers have the potential to manipulate analysis results, posing a considerable threat to people's well-being in the context of connected living.

B. High-Demand Lightweight AI Model

Researchers from both industry and academia share a common aspiration: to empower IoT devices with numerous real-time AI functions and services, with the aim of enhancing data processing and analysis within the realm of connected living. However, it's essential to acknowledge that the potential for performance improvement in IoT devices, particularly in terms of computing capabilities, is inherently constrained by factors like their small physical sizes and limited battery life. In light of these constraints, an alternative approach is to develop and deploy lightweight AI models, such as lightweight CNN models for tasks like ECG signal detection, so as to adapt to the variation in the computing capabilities of IoT devices.

C. Testbed Construction

Practically implementing strategies and algorithms for task offloading and data analysis in connected living, including the presented DRL-based framework in this article, requires the deployment of distributed computing infrastructures across different entities within the computing continuum. It's essential to consider the associated costs of deploying, operating, and maintaining these infrastructures. Despite the considerable progress achieved thus far, there remains a long way to go before achieving practical real-world application. Accordingly, an alternative way involves designing a testbed that enables the exploration and evaluation of the presented algorithms and strategies in a more flexible and cost-effective manner.

VI. Conclusion

In the connected living environment, people use a wide variety of connected devices for purposes ranging from entertainment and health management to social interaction. In this article, we discussed task offloading and data analysis in connected living, leveraging an edge-to-cloud continuum approach. More specifically, we utilize IoT devices with somewhat over-provisioned resources to provide computing services in the connected living environment. A DRL-based framework is designed to determine where to undertake the computation for connected living. Furthermore, it elaborates on the collaborative dynamics inherent in the computing continuum, encompassing both cloud-based model development and edge-based inference processes. Additionally, we conscientiously discuss certain challenges inherent in this domain, to stimulate further interest and exploration in this exciting field.

References

[1] L. Verde, N. Brancati, G. D. Pietro, M. Frucci, and G. Sannino, "A deep learning approach for voice disorder detection for smart connected living environments," *ACM Trans. Internet Techn.*, vol. 22, no. 1, pp. 8:1–8:16, 2022.

[2] Z. Lv, L. Qiao, and S. Verma, Kavita, "AI-enabled IoT-edge data analytics for connected living," *ACM Trans. Internet Techn.*, vol. 21, no. 4, pp. 104:1–104:20, 2021.

[3] H. Huang, S. Hu, and Y. Sun, "Energy-efficient ECG signal compression for user data input in cyber-physical systems by leveraging empirical mode decomposition," *ACM Trans. Cyber Phys. Syst*, vol. 3, no. 4, pp. 40:1–40:19, 2019.

[4] M. S. Rahman, I. Khalil, X. Yi, M. Atiquzzaman, and E. Bertino, "A lossless data-hiding based IoT data authenticity model in edge-ai for connected living," *ACM Trans. Internet Techn.*, vol. 22, no. 3, pp. 57:1–57:25, 2022.

[5] S. J. S. Moe et al., "Collaborative worker safety prediction mechanism using federated learning assisted edge intelligence in outdoor construction environment," *IEEE Access*, vol. 11, pp. 109010–109026, 2023.

[6] C. K. Dehury, S. N. Srirama, P. K. Donta, and S. Dustdar, "Securing clustered edge intelligence with blockchain," *IEEE Consum. Electron. Mag.*, vol. 13, no. 1, pp. 22–29, Jan. 2024.

[7] S. Duan et al., "Distributed artificial intelligence empowered by end-edge-cloud computing: A survey," *IEEE Commun. Surv. Tut.*, vol. 25, no. 1, pp. 591–624, 1st Quart. 2023.

[8] C. Tang, W. Chen, C. Zhu, Q. Li, and H. Chen, "When cache meets vehicular edge computing: Architecture, key issues, and challenges," *IEEE Wirel. Commun.*, vol. 29, no. 4, pp. 56–62, Aug. 2022.

[9] A. Knari, M. Derfouf, M. Koulali, and A. Khoumsi, "Multi-agent deep reinforcement learning for content caching within the internet of vehicles," *Ad Hoc Netw.*, vol. 152, 2024, Art. no. 103305.

[10] Z. Ning et al., "5G-enabled UAV-to-community offloading: Joint trajectory design and task scheduling," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 11, pp. 3306–3320, Nov. 2021.

[11] X. Wang, S. Wang, Y. Wang, Z. Ning, and L. Guo, "Distributed task scheduling for wireless powered mobile edge computing: A federated-learning-enabled framework," *IEEE Netw.*, vol. 35, no. 6, pp. 27–33, Dec. 2021, doi: 10.1109/MNET.201.2100179.

[12] Z. Zhang, N. Wang, H. Wu, C. Tang, and R. Li, "MR-DRO: A fast and efficient task offloading algorithm in heterogeneous edge/cloud computing environments," *IEEE Internet Things J.*, vol. 10, no. 4, pp. 3165–3178, Feb. 2023.

[13] G. P. Koslovski, K. Pereira, and P. R. Albuquerque, "Dag-based workflows scheduling using actor-critic deep reinforcement learning," *Future Gener. Comput. Syst.*, vol. 150, pp. 354–363, Jan. 2024.

[14] C. Sun, X. Li, J. Wen, X. Wang, Z. Han, and V. C. M. Leung, "Federated deep reinforcement learning for recommendation-enabled edge caching in mobile edge-cloud computing networks," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 3, pp. 690–705, Mar. 2023.

[15] Y. Ding, W. Fang, M. Liu, M. Wang, Y. Cheng, and N. Xiong, "JMDC: A joint model and data compression system for deep neural networks collaborative computing in edge-cloud networks," *J. Parallel Distrib. Comput.*, vol. 173, pp. 83–93, Mar. 2023.

Biographies

Chaogang Tang (Member, IEEE) received the B.S. degree from Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2007, and the Ph.D. degree from the School of Information Science and Technology, University of Science and Technology of China, Hefei, China, in 2012. Currently, he is an Assistant Professor with the School of Computer Science and Technology, China University of Mining and Technology. His research interests include vehicular edge computing and internet of things.

Huaming Wu (Senior Member, IEEE) received the B.E. and M.S. degrees from Harbin Institute of Technology, China, in 2009 and 2011, respectively, and the Ph.D. (with highest Hons.) degree with Freie Universität Berlin, Germany, in 2015. Currently, he is an Associate Professor with the Center for Applied Mathematics, Tianjin University. His research interests include wireless networks, mobile edge computing, internet of things, and complex networks.

Ruidong Li (Senior Member, IEEE) received the bachelor's degree from Zhejiang University, China, in 2001, and the Ph.D. degree from the University of Tsukuba, in 2008. Currently, he is an Associate Professor with the College of Science and Engineering, Kanazawa University, Japan. His research interests include big data networking, information-centric network, network security, and quantum internet.