Qubahan Academic Journal
گوڤارا قبهان يا ئەكاديمى
مجلة قبهان الأكاديمية
Doi: 10.58429/Issn.2709-8206

A Survey on IoT Task Offloading Decisions in Multi-access Edge Computing: A Decision Content Perspective

Wang Dayong[1], Kamalrulnizam Bin Abu Bakar[2], Babangida Isyaku[3]
[1,2]Faculty of Computing, Universiti Teknologi Malaysia, Malaysia
[3]Faculty of Information Communication Technology, Sule Lamido University, Nigeria
https://doi.org/10.58429/qaj.v3n4a220

Abstract
The rapid development of Internet of Things (IoT) technologies has led to increasingly complex software systems on Terminal Devices (TDs). This increases the computational load and battery consumption of TDs. The emergence of Multi-access Edge Computing (MEC) and computing offloading technology allows TDs to delegate computing-intensive tasks to the MEC network for remote execution. However, the computing and communication resources of MEC networks are limited and heterogeneous. In addition, some TDs may have a higher mobility. Therefore, IoT networks need to dynamically decide to offload some or all of the computational tasks to appropriate nodes in the MEC network. Existing reviews do not fully cover the multiple decision-making content of task offloading, and some studies do not clearly define the boundary between task offloading decision-making and task offloading scheduling optimization. This study investigates the similarities and differences between the enabling technologies, deployment architectures, and decision items of various decision mechanisms from the perspective of offloading decision content. Thus, the development and existing challenges of task offloading decision-making methods are comprehensively demonstrated, and future research directions are proposed for IoT task offloading decision-making in MEC.

Keywords: Decision-making, Edge Computing, IoT, Multi-objective Optimization, Task Offloading

1. Introduction

The development of new functions and features has caused the software system on the terminal device to become more and more complex. Complex software functions will generate a large number of computing tasks on the TD and consume more battery power of the TD (Bellini et al., 2022). However, TD's computing power and battery capacity are relatively limited. This results in slow response times of software programs and reduced battery life, which severely damages system performance and reliability (Bakar et al., 2023; Q. Li et al., 2019).

The emergence of task offloading technology provides a new optimization method to solve such contradictory problems. It allows TD to offload part or all of its computing tasks to surrounding network resources, thereby reducing the consumption of computing resources and energy consumption of TD itself. (Jaddoa et al., 2020). However, heterogeneous MEC computing networks also have resource constraints. In order to effectively offload computing tasks to distributed edge computing resources, offloading decisions need to be made based on the available status of network resources and the diverse execution requirements of tasks (Xiao et al., 2020). Many existing task offloading decision-making methods do not fully discussed multiple aspects of the decision items and lack consideration of dynamic changes in the network environment (Isyaku et al., 2022). This makes it difficult to generate an optimal solution for offloading decisions.

While some studies survey MEC enabled task offloading optimization, they mainly focus on high level scheduling that involves offloading decisions and resource allocation (Abu-Taleb et al., 2022;

Shakarami, Ghobaei-Arani, & Shahidinejad, 2020). In such papers, the concepts of task offloading decision, task offloading scheduling, task allocation and resource allocation are not defined uniformly (H. Jin et al., 2022). In addition, many studies have not fully discussed the multiple sub-problems involved in decision offloading (Gupta et al., 2022; Islam et al., 2021; Maray & Shuja, 2022; Saeik et al., 2021; Shakarami, Ghobaei-Arani, Masdari, et al., 2020). Although some studies focus on computing task offloading decisions, but they do not take into account the particularities of MEC environments (Alizadeh et al., 2020; Kaur et al., 2021). Table 1 shows related research involving task offloading decisions, as well as their focus and weaknesses. This study precisely focuses on the offloading decision-making methods and mechanisms in the MEC environment based on the latest research results of computing task offloading, and proposes a decision-item based task offloading decision-making classification.

Table 1. Related survey on task offloading scheduling

Reference	Proposed	Classification	Weaknesses
(Abu-Taleb et al., 2022)	Methods and models related to offloading decisions are discussed	Survey	Lack to distinguish between mandate allocation and resource allocation
(Shakarami, Ghobaei-Arani, & Shahidinejad, 2020)	Reviewed machine learning based optimization methods for task offloading	Survey	Differences between control objects for offloading decisions are ignored
(H. Jin et al., 2022)	Reviewed key issues, metrics, and future directions for smart computing offloading	Review	Lack to distinguish between mandate allocation and resource allocation
(Maray & Shuja, 2022)	Classification of algorithm-based techniques for task offloading in MCC and MEC was studied	Survey	Ignored task slicing and parallel offloading
(Saeik et al., 2021)	Mathematics, Artificial Intelligence and Control Theory Solutions	Survey	Task offloading between TDs is ignored
(Gupta et al., 2022)	Emphasis is placed on optimization of data transfer and storage	Review	Lack to review of the decision-making mechanisms details
(Islam et al., 2021)	Different ways of driving offload decisions are reviewed	Survey	Incomplete discussion of multiple aspects of offloading decisions
(Shakarami, Ghobaei-Arani, Masdari, et al., 2020)	Optimization techniques for task offload scheduling were studied	Systematic Review	Lack of in-depth discussion of offloading decisions
(Kaur et al., 2021)	Discussed the classification of technical routes for task offloading decisions	Systematic Review	The specificity of the MEC environment is ignored
(Alizadeh et al., 2020)	Optimization techniques for task offloading decisions are studied at the fog computing level	Systematic Review	Task offloading decision-making features does not focus on MEC

Briefly, the above-mentioned studies have some weak points as follows:

- Some of studies does not clearly distinguish between task offloading decisions and scheduling optimization
- Some of studies does not discussed in depth the correlation and differences between task offloading and resource allocation

- Some of studies does not comprehensively review the multiple decision items involved in task offloading decisions
- Some of studies discussed task offloading decisions only at the fog-computing and edge-computing level without examining the specificities of the MEC environment

The above-mentioned weakness was the lack of a dedicated research perspective on offloading decision-making problems to organize a review paper on IoT task offloading decisions in MEC environment to highlight all of these issues and future research directions. This paper comprehensively reviews the latest research on task offloading decision-making and computation offloading optimization in the MEC environment. The decision item differences covered by different task offloading decision methods are analyzed, and a taxonomy of computational task offloading decision methods based on decision item differences are proposed. Furthermore, we analyze the support relationship between different kinds of decision algorithm positions, architectures, and decision items, and give some open research challenges and future research directions.

The rest of the paper is organized as follows: Section II presents an overview description of IoT task offloading architecture and challenges. Section III discuss the classification of IoT task offloading decisions. Sections IV gives the results and discussion of analysis of several offloading decision-making methods. Section V describes open issues and future research directions. In the end, the conclusion is provided in section VI.

2. Architecture and Challenges of IoT Task Offloading

There are many architectures for offloading IoT computing tasks. In general, TDs can offload computing tasks to the surrounding computing network or other TDs. In the vertical direction, cloud computing platforms, fog computing networks, and MEC networks can all carry computing tasks offloaded by TD (X. Jin et al., 2022). In the horizontal direction, TDs can offload computing tasks from each other, thereby making full use of the currently idle available resources of other TDs (Ahmed et al., 2022). Figure 1 shows the overall architecture of IoT computing task offloading network.

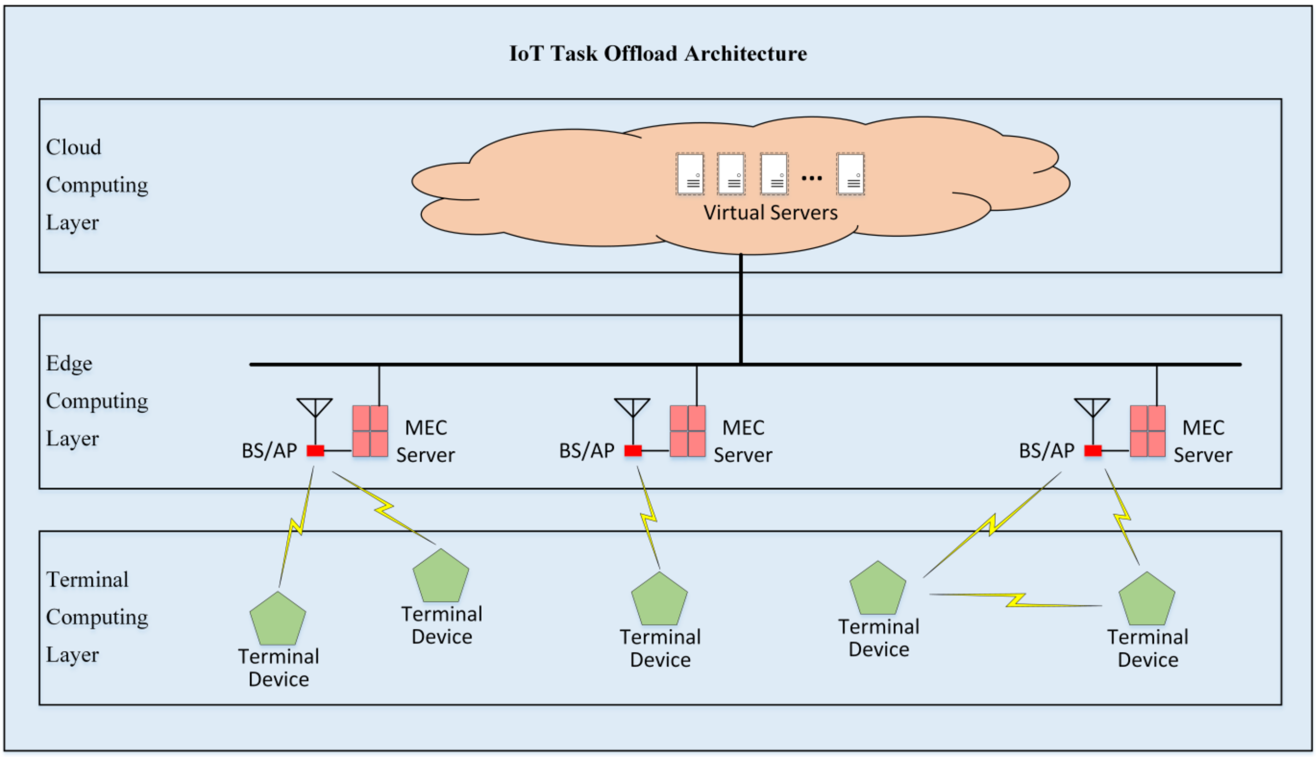


Figure 1. IoT task offload architecture

In the terminal computing layer, the amount of task offloading between TDs will not be very large, because the resources of other TDs are not very abundant (Jaddoa et al., 2020). Due to the limited mobility and wireless communication range of TDs, task offloading between TDs is highly random.

Therefore, the Terminal Computing Layer usually does not adopt a centralized task offloading decision-making scheme. Instead, the offloading decision algorithm runs in a distributed manner on each TD participating in task offloading.

The edge computing layer is the main area where computing tasks are offloaded. A large number of heterogeneous computing nodes are distributed in this layer (Bakar et al., 2023). In a real network environment, the computing resources of this layer may consist of edge computing resources co-located with base station (BS) and fog computing nodes distributed in multiple fixed locations on the network. In order to support different application scenarios, the task offloading decision-making methods in the edge computing layer have developed into two types: centralized and distributed.

The cloud platform is only used to handle large-scale computing tasks, because it is far away from TD on the network, which will introduce more communication delays (Bellini et al., 2022). Generally, only computing tasks that cannot be effectively executed by the edge computing layer will be forwarded to the cloud platform for processing. Since the cloud platform has sufficient computing resources, task offloading decisions at this level only need to consider the allocation of virtual machines (VMs). A typical application scenario for this type of task offloading is mobile cloud computing (MCC) (Saeik et al., 2021).

In order to support different application scenarios, the above multiple tasks offloading decision-making mechanisms can also cooperate jointly (Shakarami, Ghobaei-Arani, Masdari, et al., 2020). However, they all take reducing task execution time and energy consumption as the main optimization goal. In addition, the task offloading decision-making algorithm also considers the completion rate of task offloading, the battery working time of the device, and the cost of communication and other optimization goals, thus to comprehensively improve the overall performance of task offloading. Table 2 shows the main goals of task offloading.

Table 2. Main goals of task offloading decision-making

Goals	Description
Minimize Energy Consumption of TD	Reduce TD's computing energy consumption by effective tasks offloading
Minimize Computation Delay	Complete the calculation process of the task quickly through task scheduling
Maximize Reliability	Avoid task offload failure based on task migration and redundant calculation
Maximize Accuracy	Avoid inefficient task offload scheduling by dynamic decision making
Maximize Battery Life of Device	Jointly reduce computing and communication energy consumption
Minimize Communication Costs	Reduce business costs of data communications
Minimize Task Timeout	Avoid task execution timeout based on reasonable resource allocation
Maximize Task Completion Rate	Complete the task within the constrained time
Minimize Computing Cost	Comprehensive reduction of business costs for task execution

There are many factors that affect the efficiency of task offloading (Gupta et al., 2022). Therefore, the process of task offloading decision-making is complex. First, task types from different TD offloads are diverse. These tasks can only be performed on MEC nodes that partially meet the requirements, and they cannot be freely distributed to any idle nodes in the network. Second, the Quality of Service (QoS) requirements of different computing tasks are highly differentiated. Third, the available resource status of each node in the MEC network changes dynamically. In addition, the task offloading decision algorithm also needs to consider multiple influencing factors such as the mobility of the device, the commercial cost of wireless communication, and the response to task offloading failure. Table 3 shows the multiple factors affecting the generation of IoT task offloading decisions.

Table 3. Main factors of task offloading decision-making

Factors	Description
Latency	Including the calculation time and data transmission time of the task
Mobility	TDs move in MEC network environment containing multiple BS/APs
Cost	Including energy consumption, communication cost, and resource usage fees
Energy	Equipment power consumption
Load Balancing	Reasonable distribution of computing load among multiple computing nodes
Resource Constraints	Limitations of battery capacity, wireless network bandwidth and computing resources
Fault Tolerance	Tolerance for task offloading failures
Security	Controllability and integrity guarantee of task offloading process
Privacy	Prevent data from being snooped during task offloading process

Based on the above analysis, there are many issues affecting performance in the process of IoT task offloading due to the influence of many factors shown in Table 2. These issues exist in many specific domains such as task offloading decisions, task scheduling, resource allocation, and task migration.

- Cost Estimation for Task Execution. Estimating the completion time of a task is very important for deciding where to assign the task to be executed because the task offload scheduling algorithm needs to generate decisions for task offload resource allocation based on the estimated completion time of the task. Thus, it is necessary to consider the execution time of the task on the TD and the execution time of the task on the edge-computing network for efficient task offloading. In addition, the energy consumption of task execution and data transfer needs to be considered.
- Resources Allocation. Unlike the cloud-computing platform, the computing resources of the MEC network are limited. How to effectively allocate appropriate resources for tasks to be offloaded is a major challenge to improve the efficiency of task offloading.
- Task Scheduling. A large number of offloading tasks cannot be started at the same time due to computational resource limitations. How to arrange the execution order of tasks according to the various requirements of tasks and available computing resources is an important challenge of task offloading.
- Dynamic Decision Making. Task offloading decisions are complex due to the large number of factors that affect task offload performance. Such multi-objective optimization problems have been proven to be a kind of NP-Hard problems. How to comprehensively consider these influencing factors and try to achieve multi-objective optimization is an urgent problem to be solved in task offloading scheduling.
- Task Migration. The assigned tasks may need to be migrated from the current compute node to a new compute node due to the rapid movement of TD's and the change in the state of available resources in the MEC environment.

According to the previous discussion, the comprehensive optimization of scheduling for IoT task offload needs to be thoroughly investigated.

3. Classification of IoT Task Offloading Decisions

The development of new generation communication technologies such as 5G provides higher bandwidth and lower communication delay for wireless data communication. Therefore, the application of MEC has been developed rapidly. Compared with Cloudlet and traditional fog-computing, MEC can support higher device mobility and higher-performance computing task offloading services. However, it is still difficult to make complex task offloading decisions on large-scale MEC.

Computational task offloading requires decision-making content that includes complex items in order to achieve comprehensive performance optimization. However, existing research does not fully cover all decision-making content. Based on the research on the state-of-the-art offloading decision mechanism, this paper proposes a classification of offloading decisions based on decision items. Figure 2 shows the classification of IoT task offloading decisions based on decision items.

- Local/Remote computation decision: Determine whether to offload the task to the remote node or execute it locally in TD based on the task's offloading revenue prediction and available resource status. Reference (Han et al., 2023) implements the basic decision-making method of whether a task should be offloaded, which is only used to determine whether a task should be executed locally based on the expected benefits of task offloading. However, this decision content is usually implemented in conjunction with other decision items. The method proposed by (Zheng et al., 2023) (Xie et al., 2023) (D. Wang et al., 2023) simultaneously solves whether the task should be offloaded and the remote task execution node.
- Node allocation decision is responsible for selecting the appropriate remote execution node for multiple tasks to be offloaded according to the existing task offloading requirements and the workload status of the target computing network (Gao et al., 2023; K. Li et al., 2023; Sun et al., 2023).
- Offload sequence decision is responsible for scheduling the start time and sequence of multiple offload tasks according to the principle of revenue maximization (Hsieh et al., 2023; Y. Li, Zhu, Song, et al., 2023).
- Task splitting decisions: Determine whether to split the task into multiple parts and how to split it based on the dependency logic within the human flow (Song et al., 2022; Trinh & Muntean, 2023; J. Wang et al., 2022).
- Parallel offload decision solves the problem of distributed parallel execution optimization of divisible tasks (Hsieh et al., 2023).
- Resource allocation decisions of computing determine how to allocate appropriate computing resources to each task to be offloaded (Gao et al., 2023; Sun et al., 2023; You et al., 2022).
- Resource allocation decisions of communication decide to allocate network communication resources for each task to be offloaded, such as: communication time slot, signal strength, available bandwidth, etc. (Chu et al., 2023; Song et al., 2023; Ye et al., 2022)
- Hybrid decision: Jointly consider multiple offloading decisions discussed above to achieve broad task execution performance improvement (Fan et al., 2022).

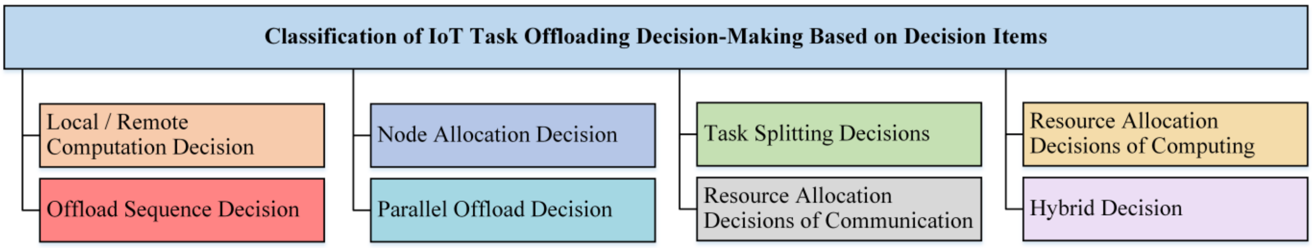


Figure 2. Classification of IoT task offloading decision-making based on decision items

In order to optimize task offloading decisions in complex network environments, researchers try to arrange offloading decision-making algorithms at multiple locations in the network. Some of these algorithms are designed to run on cloud platforms. The offloading decision-making running on Cloud Server grasps the available status of computing resources in multiple regions of a large-scale MEC network, and is conducive to making full use of sufficient computing resources to execute large-scale and complex task offloading optimization algorithms. However, the communication delay added by long-distance communication is not conducive to a quick and timely response to TD's task offloading request. Thus, the offloading decision-making method deployed on the cloud server is more suitable for supporting TD mobility in a large area (Fan et al., 2022; He et al., 2023; Song et al., 2022; Trinh &

Muntean, 2023; Ye et al., 2022; You et al., 2022). In contrast, decision-making algorithms can also be deployed directly on TDs. The offloading decision-making running on TD is conducive to quickly realizing task offloading negotiation between TDs in an environment with unstable communication quality. However, this may burden TD with repeated information exchanges (Deng et al., 2023; Han et al., 2023; K. Li et al., 2023; Y. Li, Zhu, Song, et al., 2023; D. Wang et al., 2023; Xie et al., 2023). In addition, the offloading decision can also be deployed on the edge server. In addition, multi-layer deployment is possible at the edge layer. The offload decision running on the Edge Server is conducive to fast data communication with TD. Compared with the offloading decision-making scheme deployed on TD, the offloading decision-making mechanism deployed on the edge server grasps more complete information on task offloading requirements and available computing resources in the region (Akter et al., 2023; Chu et al., 2023; Song et al., 2023; Sun et al., 2023; Yang et al., 2022; Zheng et al., 2023). In order to achieve better global task offloading scheduling, the offloading decision algorithm can also be deployed in multiple network layers and rows at the same time. The Multi-layer Federation deployment method can strike a balance between fast offloading request response and global optimization capabilities. In addition, this deployment scheme can also better support the offloading of computing tasks between multiple MEC and Fog network layers (Gao et al., 2023; Hsieh et al., 2023; Y. Li, Zhu, Li, et al., 2023; J. Wang et al., 2022). Figure 3 shows the Classification of IoT Task Offloading Decisions Based on Decision Algorithm Location.

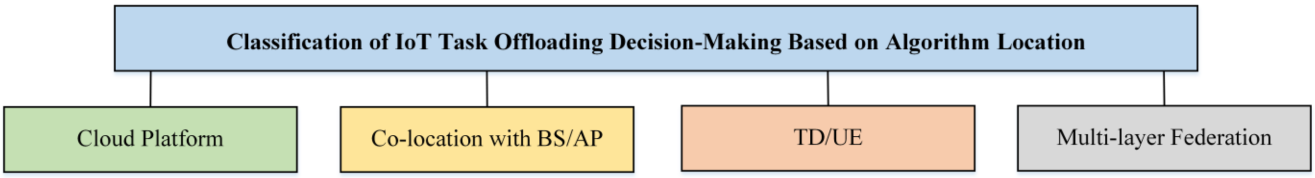


Figure 3. Classification of IoT Task Offloading Decision-making Based on Algorithm Location

Centralized decision-making is conducive to fully grasping the task requirements to be offloaded and the available MEC resources. This is conducive to the allocation of computing and communication resources, and can better realize parallel task offloading in the MEC with abundant computing resources. In addition, TD only needs to communicate with the decision controller, thus avoiding a large number of many-to-many information exchanges between TDs (Akter et al., 2023; He et al., 2023; Sun et al., 2023; Trinh & Muntean, 2023; You et al., 2022; Zheng et al., 2023). Distributed decision-making is beneficial to reduce the computing pressure of the controller, and is more suitable for working in a network environment with unstable communication quality. In addition, distributed decision making is more suitable to support task offloading between TDs. Therefore, distributed decision-making has better support for the realization of decision-making content such as task segmentation and negotiation of task offloading order (K. Li et al., 2023; Y. Li, Zhu, Li, et al., 2023; Y. Li, Zhu, Song, et al., 2023; Song et al., 2023; D. Wang et al., 2023; Xie et al., 2023).

The traditional mathematical algorithm is the basic method to realize the decision-making of task offloading. Y. Li et al. (Y. Li, Zhu, Song, et al., 2023) implements task offloading decision-making based on integer linear programming. This type of method works more efficiently in simple decision-making scenarios. However, the performance of such algorithms decreases significantly as the complexity of the decision content increases. B. Han et al. (Han et al., 2023) proposed a task offloading decision based on Impatient Queuing, allowing TD to cancel task offloading requests from different queues and dynamically choose to join the queue with a shorter waiting time. Compared with traditional methods, task offloading decision-making based on heuristic algorithms can solve relatively complex multi-objective and multi-condition constraint decision-making problems. Thus, this facilitates the implementation of joint optimization of multi-decision content (Chu et al., 2023; Song et al., 2022, 2023). Song S et al. (Ye et al., 2022) allocates resources for TD offloaded tasks based on evolutionary algorithms. S. Akter et al. (Akter et al., 2023) realizes UAV multi-objective task offloading decision-making based on genetic algorithms, and performs comprehensive optimization between UAV energy consumption and multi-task constraint objectives. Li Y et al. (Y. Li, Zhu, Li, et al., 2023) implements computing task offloading optimization that supports multi-tier deployment based on bee swarms. Decision-making based on game theory is suitable for distributed offloading decision-making, which can better support the decision-making of task computing node selection, task division and task offloading order (He et al., 2023; Hsieh et al., 2023). F. You et al. (You et al., 2022) jointly allocates computing resources and communication resources to offloaded tasks based on multi-layer games. Y. Deng et al. (Deng et al., 2023) implements multi-hop task offloading decision optimization based on classic game theory. Machine learning-based decision-making enables automatic optimization of task offloading. Supervised and unsupervised machine learning technology can avoid pre-designing complex decision-making recursive processes. On the contrary, the AI model can learn the requirements for task offloading and the characteristics of network resource status changes by itself, and then automatically give optimized offloading decisions (D. Wang et al., 2023; Xie et al., 2023). In addition, the reinforcement learning method can automatically learn the task offloading requirements and the state changes of the available resources in the MEC network, so as to gradually grasp the law of optimal matching between the two (Fan et al., 2022; Gao et al., 2023). Compared with machine learning, more complex logical relationships can be identified based on deep learning models. Therefore, task offloading decision-making based on deep learning can support massive constraint conditions and system index learning, thereby realizing extremely complex task offloading decisions (Sun et al., 2023; Trinh & Muntean, 2023; J. Wang et al., 2022; Zheng et al., 2023). K. Li et al. (K. Li et al., 2023) implements self-learning task offloading decisions between TDs based on deep reinforcement learning. The method proposed by S. Yang et al.(Yang et al., 2022) can quickly allocate computing and communication resources to dynamically changing offloading tasks based on deep learning. Wang J et al. (J. Wang et al., 2022) proposed a task offloading decision framework based on deep reinforcement learning to support unloading task cutting and partial offloading.

Table 4. Comparison of Individual Studies Addressing Task Offloading Decision

Works	Contribution	Decision Items	Techniques	Location	Schema	Optimization Goals
(Han et al., 2023)	Task offloading decision with TD cancellation mechanism	Local / Remote;	Impatience based Queuing	TD	Distributed	Maximize Reliability; Minimize Computation Delay;
(Zheng et al., 2023)	Wireless-powered TD task offloading decision-making method in EC with single access point	Local / Remote; Node Allocation	DRL	Edge Server	Centralized	Minimize Computation Delay
(Xie et al., 2023)	Distributed TD-to-TD offloading decision based on game theory	Local / Remote; Node Allocation	Learning-theoretic-based Algorithm	TD	Distributed	Maximize Network-wide Utility
(D. Wang et al., 2023)	Distributed task offloading decisions in large-scale MEC environments	Local / Remote; Node Allocation	Generative Adversarial Networks (GAN)	TD	Distributed	Minimize Computation Delay; Minimize Energy Consumption
(K. Li et al., 2023)	TD-to-TD task offloading mechanism based on attention communication deep reinforcement learning	Local / Remote; Node Allocation	DRL	TD	Distributed	Minimize Computation Delay; Maximize Task Completion Rate

(Gao et al., 2023)	A new approach to integrating centralized training and distributed decision making	Computing Resource Allocation	Multi-Agent Reinforcement Learning	Multi-layer Federation	Distributed	Maximize Task Completion Rate; Minimize Computing Cost
(Sun et al., 2023)	A graph reinforcement learning-based offloading framework	Local / Remote; Computing Resource Allocation	Graph-RL	Edge Server	Centralized	Minimize Computation Delay
(Hsieh et al., 2023)	Task offloading method based on Knapsack Potential Game (KPG)	Node Allocation; Parallel Offload; Offload Sequence	KPG	Multi-layer Federation	Distributed	Minimize Computation Delay
(Y. Li, Zhu, Song, et al., 2023)	Efficient Offload Decisions for Mobile Augmented Reality (MAR)	Offload Sequence; Node Allocation	Integer Nonlinear Programming	TD	Distributed	Minimize Energy Consumption; Maximize Accuracy
(Song et al., 2022)	Computing offloading decision optimization method considering computing node load balancing	Task Splitting; Node Allocation	Ant Colony Optimization	Cloud Server	Centralized	Minimize Computation Delay;
(Trinh & Muntean, 2023)	VR task offloading decision-making method with task splitting	Local / Remote; Node Allocation; Task Splitting	DRL	Cloud Server	Centralized	Minimize Computation Delay; Minimize Energy Consumption
(J. Wang et al., 2022)	Proposed a framework to support partial computation offloading with channel utilization optimization	Task Splitting; Node Allocation	DRL	Multi-layer Federation	Distributed	Minimize Computing Cost; Minimize Energy Consumption
(You et al., 2022)	Task offloading optimization mechanism with cloud-edge collaboration	Computing and Communication Resource Allocation	Multi-layer Game-theoretic	Cloud Server	Centralized	Maximize Task Completion Rate; Minimize Computing Cost

(Ye et al., 2022)	Balanced offloading strategy for IoT	Node Allocation; Communication Resource Allocation	Evolutionary Algorithm	Cloud Server	Centralized	Minimize Computation Delay; Minimize Energy Consumption
(Song et al., 2023)	Joint bandwidth allocation and task offloading based on heuristic algorithms	Local / Remote; Communication Resource Allocation	PSO	Edge Server	Distributed	Minimize Computation and Communication Delay
(Chu et al., 2023)	Online task offloading decision-making mechanism with cost awareness	Computing and Communication Resource Allocation	Heuristic	Edge Server	Distributed	Minimize Computation Delay; Minimize Communication Costs
(Fan et al., 2022)	A computational offloading decision-making method with multi-objective optimization in rapidly changing MEC environments	Local / Remote; Node Allocation; Communication Resource Allocation	RL	Cloud Server	Centralized	Minimize Computation Delay
(He et al., 2023)	Task offloading decision-making in cloud-assisted MEC	Local / Remote; Node Allocation	Karush–Kuhn–Tucker (KKT)	Cloud Server	Centralized	Minimize Energy Consumption; Maximize Task Completion Rate
(Deng et al., 2023)	Multi-hop enabled MEC task offloading decision-making scheme·	Node Allocation	Game Theoretic	TD	Distributed	Maximize Task Completion Rate; Minimize Task Timeout
(Yang et al., 2022)	An offloading decision-making method that quickly adapts to dynamically changing MEC scenarios	Computing and Communication Resource Allocation	Deep Supervised Learning	Edge Server	Centralized	Minimize Computation Delay; Minimize Energy Consumption
(Akter et al., 2023)	A dynamic offloading method for computational tasks in multilayer UAV networks	Local / Remote; Node Allocation	Messy Genetic Algorithm	Edge Server	Centralized	Maximize Battery Life of Device; Minimize Computation Delay;

(Y. Li, Zhu, Li, et al., 2023)	MAR's task offloading decision method with task cache	Resource Allocation; Node allocation	Artificial Bee Colony	Multi-layer Federation	Distributed	Minimize Computation Delay

4. Results and Discussions

According to the previous discussion, it was found that existing task offloading decision methods rarely comprehensively cover the complete decision function. Some algorithms only address part of the function in task offloading decisions. In general, most task offloading methods can decide whether a task should be offloaded, and about 50% of the methods can select the appropriate task execution node in MEC for the offloaded task. However, only a few offloading decision-making methods for specific application scenarios support advanced offloading decision-making functions such as task segmentation, parallel task offloading, and task migration. Figure 4 shows the decision-making items included in the investigated offloading decision methods.

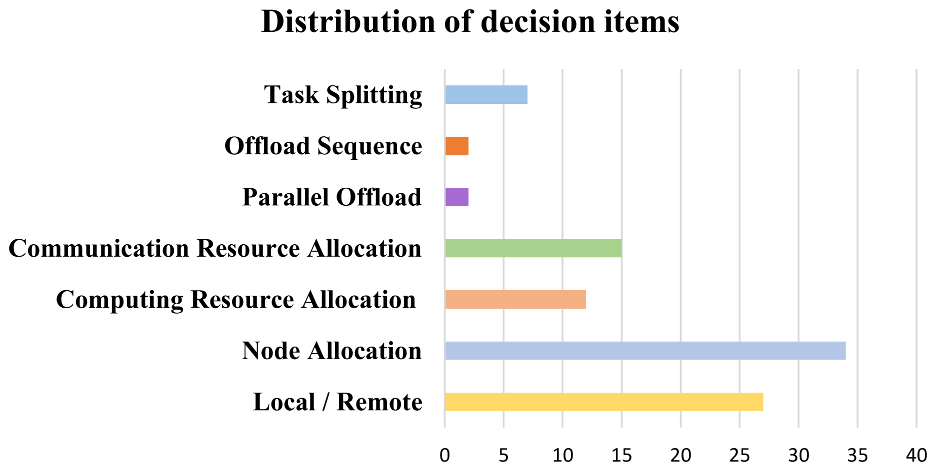


Figure 4. Distribution of decision item

The centralized task offloading decision-making architecture is more conducive to achieving global task offloading optimization within the service area. However, this architecture is not conducive to handling load balancing and single point of failure issues. The distributed task offloading decision-making architecture is conducive to supporting horizontal task offloading between TDs. However, there is a disadvantage of large interactive communication volume. The offloading decision-making technology of both schemes is still developing due to different application scenario requirements. Figure 4b shows the distribution of the two schemes involved in this survey.

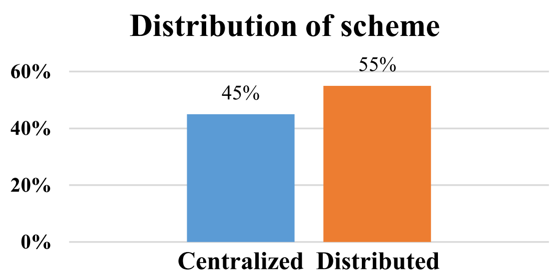
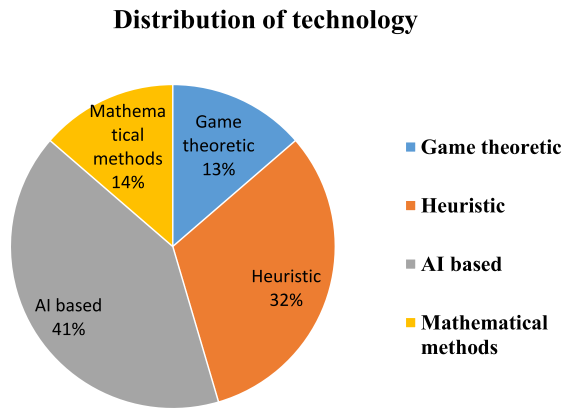
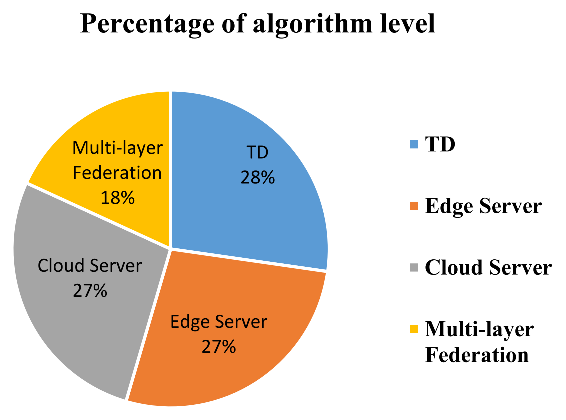


Figure 5. Classification of scheme

Task offloading methods based on machine learning and deep learning have developed rapidly in recent years, and they can support more complex application scenarios and comprehensive optimization of more goals. However, offloading decision-making algorithms based on AI technology usually incur a large computational cost, which makes it unsuitable to run on performance-limited TDs. In contrast, game theory-based decision-making methods have advantages in distributed decision-making architectures that support D2D task offloading. In addition, decision-making methods based on mathematical optimization can provide high cost performance in relatively fixed and simple network application scenarios. Figure 6a shows the proportion of task offloading technology solutions.



(a)



(b)

Figure 6. (a).Distribution of technology, (b). Percentage of algorithm resides level

Task offloading methods based on machine learning and deep learning have developed rapidly in recent years, and they can support more complex application scenarios and comprehensive optimization of more goals. Based on the differences in application scenarios, the computing task offloading decision algorithm can be deployed on cloud platforms, edge servers and TD. However, the offloading decision algorithm in the MEC environment is more suitable to be deployed on the edge server, and jointly deployed on the edge server and TD. Figure 6b shows the distribution proportion of offloading decision algorithms residing at different network levels.

In summary, most of the existing computing task offloading decision-making algorithms are biased towards some specific optimization goals based on considering limited influencing factors. However, there is a lack of research on offloading optimization methods that comprehensively consider decision content.

5. Open Issues and Future Research Directions

Due to the development of IoT application requirements, the number and mobility of TDs have become a rapidly rising trend, which leads to frequent cross-regional task offloading. However, existing offloading decision methods lack support for highly dynamic task offloading in large-scale MEC environments. Thus, there is a need for better research intelligent offloading decision-making mechanism for highly dynamic interaction in large-scale IoT networks as follows:

Decision-making for high-density task offloading

The number of IoT TDs is also increasing rapidly with the increasing demand for service functions, this results in an increasing density of TD within unit physical space. Therefore, task offloading activities in the MEC environment will become very frequent. However, the task offloading decision-making mechanism always tends to comprehensively analyze all task offloading requirements and available computing resource status in the current time slot, which will lead to overly complex decision calculations. In addition, some decision-making algorithms have difficulty providing fast decision-response services due to the computational complexity of the algorithm itself. Moreover, there may be multiple MEC wireless access points in the same physical area. TDs may be highly dynamic in submitting task offload requests to multiple MEC access points due to signal quality fluctuations in the wireless communication network, and this further increases the difficulty of making quick offload decisions. Therefore, research on efficient computing offloading decision-making mechanisms is very necessary in high-density task offloading scenarios.

Integrated decision-making mechanism

The task offloading decision-making mechanism involves local computing judgment, remote computing node selection, communication resource allocation, and task offloading and splitting.

However, most of the existing task offloading decision-making research only optimizes certain local problems in multiple decision-making content. In fact, multiple aspects involved in computing offloading decisions affect each other, and simply optimizing one aspect of the decision-making item may lead to a significant decline in the performance of other aspects of decision-making indicators. Therefore, it is necessary to conduct in-depth research on the integrated decision-making mechanism.

Decentralized decision-making in Large-scale MEC

The centralized computing offloading decision-making mechanism is difficult to provide efficient and fast decision-making services in large-scale network environments due to its unique high computing load shortcomings. Some solutions focus on relying as much as possible on the high computing power of the cloud server to implement more complex decision-making calculations. However, cloud platforms are usually far away from TDs. Adding more additional communication delays will further reduce the responsiveness of offloading decisions. On the other hand, the communication quality between adjacent TDs may be better than the communication quality between TDs and the MEC network. Which means TDs can exchange information with each other more quickly. This way, distributed task offloading decisions may have better performance in such scenarios. Most of the existing research on distributed computing task offloading decisions assumes that the target network environment is limited to a relatively small physical area, but there is a lack of discussion on the optimization of distributed task offloading decisions in large-scale IoT networks.

Dynamic task migration

Unlike original IoT, TDs are more mobile nowadays. TDs in motion may rapidly change connections to the BS. In fact, most computing offloading decision-making mechanisms always allocate offloaded tasks to MEC computing nodes that are closer to the TD, thereby reducing the transmission delay of task data. When TDs move rapidly within the physical area, the offloading decision algorithm needs to determine whether the offloaded tasks should be migrated between different computing nodes based on the task execution progress and the movement speed of TDs. In addition, the movement trajectories of TDs are difficult to predict, and task migration will add additional costs. Therefore, dynamic computing task migration decisions need to be further studied.

High reliability and redundancy

The mobility of TDs not only increases the computational complexity of task offloading decisions, but also leads to more communication failures and timeouts. Therefore, computing task-offloading mechanisms that support mobile application scenarios need to consider redundant resource allocation to achieve higher reliability. Some studies have proposed reliability enhancement schemes based on TDs task offloading prediction. However, the trade-off between resource utilization and reliability remains difficult.

6. Conclusion

The rapid development of IoT applications has led to a continuous increase in the amount of computing tasks on terminal devices. In order to reduce the computing load and energy consumption of terminal equipment, TD tends to offload as many tasks as possible to the MEC network. However, wireless network resources and MEC computing resources are limited, so it is necessary to continuously make dynamic task offloading decisions in combination with task offloading requirements and available MEC resources, and thus to maximize the overall performance of the IoT network. This study provides a comprehensive review of the state-of-the-art research results on computational task offloading, and discusses the characteristics of various algorithms as well as their adaptability for different application scenarios. In addition, this study focuses on analyzing the multiple decision items supported by existing offloading decision-making methods and proposes a new classification of offloading decisions for computational tasks from the perspective of the algorithm's decision-making content. Furthermore, open issues and future research directions for IoT task offload decision-making in MEC environment are discussed and proposed.

References

Abu-Taleb, N. A., Hasan Abdulrazzak, F., Zahary, A. T., & Al-Mqdashi, A. M. (2022). Offloading Decision Making in Mobile Edge Computing: A Survey. *2022 2nd International Conference on Emerging Smart Technologies and Applications (eSmarTA)*, 1–8. https://doi.org/10.1109/eSmarTA56775.2022.9935407

Ahmed, M., Raza Naqvi, S. S., Mirza, M., Aziz, A., Khan, M., Khan, W. U., Li, J., & Han, Z. (2022). A survey on vehicular task offloading: Classification, issues, and challenges. *Journal of King Saud University - Computer and Information Sciences*, *34*. https://doi.org/10.1016/j.jksuci.2022.05.016

Akter, S., Kim, D.-Y., & Yoon, S. (2023). Task Offloading in Multi-Access Edge Computing Enabled UAV-Aided Emergency Response Operations. *IEEE Access*, *11*, 23167–23188. https://doi.org/10.1109/ACCESS.2023.3252575

Alizadeh, M. R., Khajehvand, V., Rahmani, A. M., & Akbari, E. (2020). Task scheduling approaches in fog computing: A systematic review. *International Journal of Communication Systems*, *33*(16), e4583. https://doi.org/10.1002/dac.4583

Bakar, K. B. A., Zuhra, F. T., Isyaku, B., & Sulaiman, S. B. (2023). A Review on the Immediate Advancement of the Internet of Things in Wireless Telecommunications. *IEEE Access*, *11*, 21020–21048. https://doi.org/10.1109/ACCESS.2023.3250466

Bellini, P., Nesi, P., & Pantaleo, G. (2022). IoT-Enabled Smart Cities: A Review of Concepts, Frameworks and Key Technologies. *Applied Sciences*, *12*(3), Article 3. https://doi.org/10.3390/app12031607

Chu, W., Yu, P., Yu, Z., Lui, J. C. S., & Lin, Y. (2023). Online Optimal Service Selection, Resource Allocation and Task Offloading for Multi-Access Edge Computing: A Utility-Based Approach. *IEEE Transactions on Mobile Computing*, *22*(7), 4150–4167. https://doi.org/10.1109/TMC.2022.3152493

Deng, Y., Chen, Z., Chen, X., & Fang, Y. (2023). Task Offloading in Multi-Hop Relay-Aided Multi-Access Edge Computing. *IEEE Transactions on Vehicular Technology*, *72*(1), 1372–1376. https://doi.org/10.1109/TVT.2022.3204398

Fan, W., Liu, J., Hua, M., Wu, F., & Liu, Y. (2022). Joint Task Offloading and Resource Allocation for Multi-Access Edge Computing Assisted by Parked and Moving Vehicles. *IEEE Transactions on Vehicular Technology*, *71*(5), 5314–5330. https://doi.org/10.1109/TVT.2022.3149937

Gao, Z., Yang, L., & Dai, Y. (2023). Large-Scale Computation Offloading Using a Multi-Agent Reinforcement Learning in Heterogeneous Multi-Access Edge Computing. *IEEE Transactions on Mobile Computing*, *22*(6), 3425–3443. https://doi.org/10.1109/TMC.2022.3141080

Gupta, P., Sharma, R., & Gupta, S. (2022). A Review on Task Offloading Mechanism for IoT Edge Fog Cloud Data Interplay. *2022 IEEE Delhi Section Conference (DELCON)*, 1–10. https://doi.org/10.1109/DELCON54057.2022.9753215

Han, B., Sciancalepore, V., Xu, Y., Feng, D., & Schotten, H. D. (2023). Impatient Queuing for Intelligent Task Offloading in Multiaccess Edge Computing. *IEEE Transactions on Wireless Communications*, *22*(1), 59–72. https://doi.org/10.1109/TWC.2022.3191287

He, Z., Xu, Y., Liu, D., Zhou, W., & Li, K. (2023). Energy-efficient computation offloading strategy with task priority in cloud assisted multi-access edge computing. *Future Generation Computer Systems*, *148*, 298–313. https://doi.org/10.1016/j.future.2023.06.014

Hsieh, C.-Y., Ren, Y., & Chen, J.-C. (2023). Edge-Cloud Offloading: Knapsack Potential Game in 5G Multi-Access Edge Computing. *IEEE Transactions on Wireless Communications*, 1–1. https://doi.org/10.1109/TWC.2023.3248270

Islam, A., Debnath, A., Ghose, M., & Chakraborty, S. (2021). A Survey on Task Offloading in Multi-access Edge Computing. *Journal of Systems Architecture*, *118*, 102225. https://doi.org/10.1016/j.sysarc.2021.102225

Isyaku, B., Bakar, K. B. A., Ghaleb, F. A., & Al-Nahari, A. (2022). Dynamic Routing and Failure Recovery Approaches for Efficient Resource Utilization in OpenFlow-SDN: A Survey. *IEEE Access*, *10*, 121791–121815. https://doi.org/10.1109/ACCESS.2022.3222849

Jaddoa, A., Sakellari, G., Panaousis, E., Loukas, G., & Sarigiannidis, P. G. (2020). Dynamic decision support for resource offloading in heterogeneous Internet of Things environments. *Simulation Modelling Practice and Theory*, *101*, 102019. https://doi.org/10.1016/j.simpat.2019.102019

Jin, H., Gregory, M. A., & Li, S. (2022). A Review of Intelligent Computation Offloading in Multiaccess Edge Computing. *IEEE Access*, *10*, 71481–71495. https://doi.org/10.1109/ACCESS.2022.3187701

Jin, X., Hua, W., Wang, Z., & Chen, Y. (2022). A survey of research on computation offloading in mobile cloud computing. *Wireless Networks*, *28*(4), 1563–1585. https://doi.org/10.1007/s11276-022-02920-2

Kaur, N., Kumar, A., & Kumar, R. (2021). A systematic review on task scheduling in Fog computing: Taxonomy, tools, challenges, and future directions. *Concurrency and Computation: Practice and Experience*, *33*(21), e6432. https://doi.org/10.1002/cpe.6432

Li, K., Wang, X., He, Q., Yang, M., Huang, M., & Dustdar, S. (2023). Task Computation Offloading for Multi-Access Edge Computing via Attention Communication Deep Reinforcement Learning. *IEEE Transactions on Services Computing*, *16*(4), 2985–2999. https://doi.org/10.1109/TSC.2022.3225473

Li, Q., Zhao, J., Gong, Y., & Zhang, Q. (2019). Energy-efficient computation offloading and resource allocation in fog computing for Internet of Everything. *China Communications*, *16*(3), 32–41. https://doi.org/10.12676/j.cc.2019.03.004

Li, Y., Zhu, X., Li, N., Wang, L., Chen, Y., Yang, F., & Zhai, L. (2023). Collaborative Content Caching and Task Offloading in Multi-Access Edge Computing. *IEEE Transactions on Vehicular Technology*, *72*(4), 5367–5372. https://doi.org/10.1109/TVT.2022.3222596

Li, Y., Zhu, X., Song, S., Ma, S., Yang, F., & Zhai, L. (2023). Task offloading and parameters optimization of MAR in multi-access edge computing. *Expert Systems with Applications*, *215*, 119379. https://doi.org/10.1016/j.eswa.2022.119379

Maray, M., & Shuja, J. (2022). Computation Offloading in Mobile Cloud Computing and Mobile Edge Computing: Survey, Taxonomy, and Open Issues. *Mobile Information Systems*, *2022*, e1121822. https://doi.org/10.1155/2022/1121822

Saeik, F., Avgeris, M., Spatharakis, D., Santi, N., Dechouniotis, D., Violos, J., Leivadeas, A., Athanasopoulos, N., Mitton, N., & Papavassiliou, S. (2021). Task offloading in Edge and Cloud Computing: A survey on mathematical, artificial intelligence and control theory solutions. *Computer Networks*, *195*, 108177. https://doi.org/10.1016/j.comnet.2021.108177

Shakarami, A., Ghobaei-Arani, M., Masdari, M., & Hosseinzadeh, M. (2020). A Survey on the Computation Offloading Approaches in Mobile Edge/Cloud Computing Environment: A Stochastic-based Perspective. *Journal of Grid Computing*, *18*(4), 639–671. https://doi.org/10.1007/s10723-020-09530-2

Shakarami, A., Ghobaei-Arani, M., & Shahidinejad, A. (2020). A survey on the computation offloading approaches in mobile edge computing: A machine learning-based perspective. *Computer Networks*, *182*, 107496. https://doi.org/10.1016/j.comnet.2020.107496

Song, S., Ma, S., Yang, L., Zhao, J., Yang, F., & Zhai, L. (2022). Delay-sensitive tasks offloading in multi-access edge computing. *Expert Systems with Applications*, *198*, 116730. https://doi.org/10.1016/j.eswa.2022.116730

Song, S., Ma, S., Zhu, X., Li, Y., Yang, F., & Zhai, L. (2023). Joint bandwidth allocation and task offloading in multi-access edge computing. *Expert Systems with Applications*, *217*, 119563. https://doi.org/10.1016/j.eswa.2023.119563

Sun, Z., Mo, Y., & Yu, C. (2023). Graph-Reinforcement-Learning-Based Task Offloading for Multiaccess Edge Computing. *IEEE Internet of Things Journal*, *10*(4), 3138–3150. https://doi.org/10.1109/JIOT.2021.3123822

Trinh, B., & Muntean, G.-M. (2023). A Deep Reinforcement Learning-Based Offloading Scheme for Multi-Access Edge Computing-Supported eXtended Reality Systems. *IEEE Transactions on Vehicular Technology*, *72*(1), 1254–1264. https://doi.org/10.1109/TVT.2022.3207692

Wang, D., Wang, W., Gao, H., Zhang, Z., & Han, Z. (2023). Delay-Optimal Computation Offloading in Large-Scale Multi-Access Edge Computing using Mean Field Game. *IEEE Transactions on Wireless Communications*, 1–1. https://doi.org/10.1109/TWC.2023.3291198

Wang, J., Ke, H., Liu, X., & Wang, H. (2022). Optimization for computational offloading in multi-access edge computing: A deep reinforcement learning scheme. *Computer Networks*, *204*, 108690. https://doi.org/10.1016/j.comnet.2021.108690

Xiao, K., Gao, Z., Shi, W., Qiu, X., Yang, Y., & Rui, L. (2020). EdgeABC: An architecture for task offloading and resource allocation in the Internet of Things. *Future Generation Computer Systems*, *107*, 498–508. https://doi.org/10.1016/j.future.2020.02.026

Xie, J., Jia, Y., Wen, W., Chen, Z., & Liang, L. (2023). Dynamic D2D Multihop Offloading in Multi-Access Edge Computing From the Perspective of Learning Theory in Games. *IEEE Transactions on Network and Service Management*, *20*(1), 305–318. https://doi.org/10.1109/TNSM.2022.3201470

Yang, S., Lee, G., & Huang, L. (2022). Deep Learning-Based Dynamic Computation Task Offloading for Mobile Edge Computing Networks. *Sensors*, *22*(11), Article 11. https://doi.org/10.3390/s22114088

Ye, D., Wang, X., & Hou, J. (2022). Balanced multi-access edge computing offloading strategy in the Internet of things scenario. *Computer Communications*, *194*, 399–410. https://doi.org/10.1016/j.comcom.2022.07.048

You, F., Ni, W., Li, J., & Jamalipour, A. (2022). New Three-Tier Game-Theoretic Approach for Computation Offloading in Multi-Access Edge Computing. *IEEE Transactions on Vehicular Technology*, *71*(9), 9817–9829. https://doi.org/10.1109/TVT.2022.3176302

Zheng, K., Jiang, G., Liu, X., Chi, K., Yao, X., & Liu, J. (2023). DRL-Based Offloading for Computation Delay Minimization in Wireless-Powered Multi-Access Edge Computing. *IEEE Transactions on Communications*, *71*(3), 1755–1770. https://doi.org/10.1109/TCOMM.2023.3237854