End-to-End Network and Computation Resource Allocation for Task Offloading in Mobile Networks

Jaechan Lee, Yumi Kim, and Haneul Ko
Dept. of Electronics and Information Convergence Engineering
Kyung Hee University
Yongin-si, Gyeonggi-do, Korea
{jaechan,yumikim1201,heko}@khu.ac.kr

***Abstract*—This paper introduces a joint network and computation resource management framework for mobile networks, aiming to minimize task completion times in offloading scenarios. A task management function (TMF) is designed to coordinate resources across network entities, optimizing both network and computation resources while ensuring tasks meet their deadlines. In addition, a low-complexity heuristic algorithm is developed. It operates with two sub-algorithms: initial resource allocation algorithm and resource adjustment algorithm.**

***Index Terms*—Resource allocation, resource management, network resource, computation resource, optimization.**

I. Introduction

The rapid evolution of mobile networks has led to the proliferation of resource-intensive applications such as augmented reality (AR), virtual reality (VR), online gaming, and AI-driven services like deep learning and edge computing. These applications are highly demanding in terms of computational power and require ultra-low latency to provide seamless and immersive user experiences. As mobile devices continue to adopt these applications, the computational limitations and battery constraints of user equipment (UE) pose significant challenges for delivering the required quality of service (QoS).

To overcome these limitations, task offloading has emerged as a key solution [1], where computationally heavy tasks are offloaded from UEs to remote servers–typically edge or cloud servers–capable of handling complex computations with higher efficiency. This offloading reduces the computational burden on UE and extends battery life, while leveraging the superior processing power of the edge or cloud servers. However, most existing solutions focus either on optimizing computational resources at the offloading server or on improving wireless transmission [2]–[5]. These approaches lack a holistic view that considers the entire data path from UE to the offloading server and back, including both wireless and wired segments of the network.

These challenges highlight the need for an integrated resource management framework that simultaneously optimizes both network and computation resources from an end-to-end perspective. This paper addresses this gap by introducing a joint network and computation resource management framework. The framework introduces a task management function (TMF), which acts as a central controller for coordinating the allocation of resources across multiple network entities. For the optimal resource allocation, we propose a heuristic algorithm. The algorithm consists of two sub-algorithms: 1) initial resource allocation algorithm; and 2) resource adjustment algorithm.

The remainder of this paper is as follows. Section II reviews the related work; Section III presents the proposed framework; Section IV explains the hueristic algorithm; and Section V concludes the paper with final remarks.

II. Related Work

Significant efforts have been made to improve the efficiency of task offloading in mobile networks [**?**], [6]–[12].

Das *et al.* [6] introduced a container instance management system that strategically chooses deployment locations for containers, factoring in latency and operating costs. Cicconetti *et al.* [7] proposed a strategy to implement serverless computing using existing edge standards. Tariq *et al.* [8] created a lightweight computing framework designed to handle various computation chains efficiently. Sarkar *et al.* [9] developed a multi-layer computational architecture, integrating fog, edge, and central cloud resources to achieve lower latency.

Wang *et al.* [10] designed a reinforcement learning-based scheduling algorithm adjusting container counts and memory settings dynamically, optimizing the trade-off between cost and performance. Gupta *et al.* [11] implemented a resource allocation scheme that focuses on enhancing user experience by reducing task completion times. Elgamal *et al.* [12] addressed cost minimization for processing tasks while keeping task completion times within target limits, proposing a heuristic that takes into account fusion, placement, and configuration aspects. Ko *et al.* [4] developed a task offloading model that selects the optimal computational power to reduce costs while keeping processing delays to a minimum.

Despite the effectiveness of these approaches, they often lack a fully integrated view that spans the complete path from UE to the offloading server and back, including both the wireless and wired segments of the network infrastructure.

III. Proposed Framework

To address the challenges of task offloading in mobile networks, we propose a joint network and computation resource management framework as shown in Figure 1. This framework aims to efficiently manage both network and computational

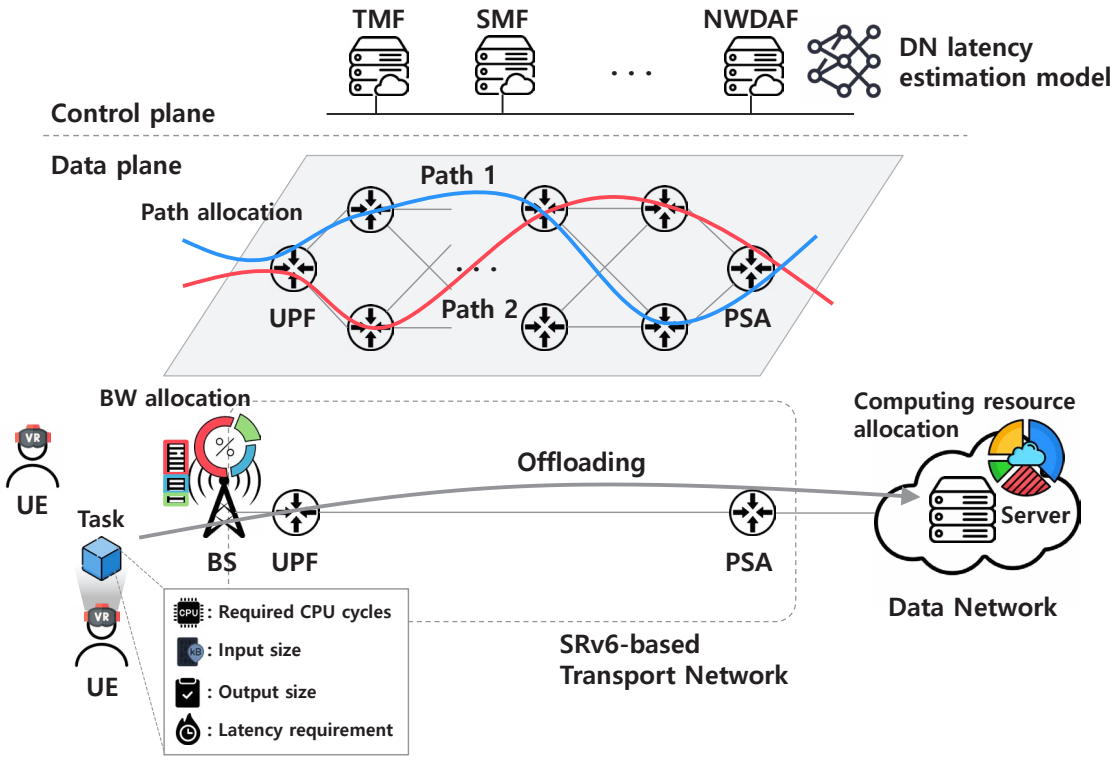

Fig. 1. Joint Network and Computation Resource Management Framework.

resources from an end-to-end perspective, covering the entire data path from UE to offloading servers. This holistic view ensures that tasks are offloaded, processed, and returned within specified deadlines, meeting the performance requirements of modern, latency-sensitive applications.

The framework is centered around TMF, a novel network entity responsible for coordinating resource allocation across multiple components in the mobile network. TMF acts as a decision-making unit that interacts with different network entities, ensuring that both network and computational resources are managed. For this, TMF's primary responsibilities include: 1) TMF gathers task information from UEs, such as deadlines, required CPU cycles, and data sizes. This information is essential for determining optimal resource allocation strategies; 2) based on the collected data, TMF determines the optimal distribution of both network and computational resources. It considers factors such as channel conditions, available bandwidth, and server processing capacity to devise an efficient strategy; and 3) TMF manages both wireless and wired segments to ensure seamless task processing.

Meanwhile, the operation procedure of the proposed framework is as follows. UE generates tasks regularly, each defined by its deadline, required CPU cycles, and the input and output data sizes. When a task is generated, UE sends an offloading request to TMF. This request includes task information along with additional network details like channel gain and transmission power. Then, TMF uses this data to periodically decide how to allocate network and computational resources for each UE. This strategy covers the allocation of uplink and downlink bandwidth portions, server computational capacity, and the selection of paths in the transport network for data transmission.

The resource allocation must ensure that each task is completed within its specified deadline. The total task completion time includes various latency components, covering data transmission times over the wireless, transport, and data networks, as well as the processing time at the offloading server. However, since the mobile network operator cannot directly control data network resources, transmission latencies in the data network are predetermined values. To ensure tasks meet their deadlines, TMF estimates these latencies by requesting analytics from the network data analytics function (NWDAF) [13]. If detailed analytics are unavailable, TMF can instead request a round trip time (RTT) measurement between the offloading server and the network's session anchor. Using the RTT data, TMF estimates the necessary transmission delays. With this information, TMF can develop a resource allocation strategy that ensures tasks are completed within their deadlines.

IV. Heuristic Algorithm

To derive the optimal resource allocation strategy in a practical manner, we develop a heuristic algorithm consisting of two sub-algorithms as follows.

Initial resource allocation algorithm: In this algorithm, network and computational resources are allocated based on task characteristics such as input/output sizes and required CPU cycles. UEs are assigned bandwidth and computation

resources proportionally, ensuring that UEs with higher resource needs receive adequate allocations. This phase sets a foundation for effective resource distribution.

Resource adjustment algorithm: After the initial allocation, UEs are categorized into "satisfied" (those meeting task deadlines) and "dissatisfied" groups. Resources from satisfied UEs are partially reallocated to dissatisfied UEs to optimize overall task completion. The algorithm iteratively adjusts the allocation until most, if not all, UEs meet their deadlines, ensuring an efficient and dynamic resource distribution.

V. Conclusion

This paper introduces a comprehensive framework for managing both network and computation resources in mobile networks, aiming to minimize task completion times during offloading scenarios. The core innovation is the task management function (TMF), which acts as a centralized decision-making entity, coordinating resource allocation across the network to ensure task deadlines are met. To achieve this, a heuristic algorithm is proposed, consisting of two key components: the initial resource allocation algorithm and the resource adjustment algorithm. The initial allocation efficiently distributes resources based on task characteristics, while the adjustment algorithm dynamically reallocates resources to tasks that need additional support to meet their deadlines. Overall, the proposed framework and heuristic algorithm provide an effective and practical solution for optimizing task offloading in time-sensitive mobile applications. In our future work, we will formulate an optimization problem for the optimal performance of the proposed framework.

Acknowledge

This research was supported in part by National Research Foundation (NRF) of Korea Grant funded by the Korean Government (MSIP) (No. RS-2024-00340698) and in part by Institute for Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2021-0-00739).

References

[1] H. Ko *et al.*, "A Belief-Based Task Offloading Algorithm in Vehicular Edge Computing," *IEEE Transactions on Intelligent Transportation Systems (T-ITS)*, vol. 24, no. 5, pp. 5467–5476, May 2023.

[2] Y. Ding *et al.*, "Online Edge Learning Offloading and Resource Management for UAV-Assisted MEC Secure Communications," *IEEE Journal of Selected Topics in Signal Processing*, vol. 17, no. 1, pp. 54–65, Jan. 2023.

[3] Y. Chen *et al.*, "Energy Efficient Dynamic Offloading in Mobile Edge Computing for Internet of Things," *IEEE Transactions on Cloud Computing (TCC)*, vol. 9, no. 3, pp. 1050–1060, Jul.-Sep. 2021.

[4] H. Ko, S. Pack, and V. Leung, "Performance Optimization of Serverless Computing for Latency-Guaranteed and Energy-Efficient Task Offloading in Energy Harvesting Industrial IoT," *IEEE Internet of Things Journal (IoT-J)*, vol. 10, no. 3, pp. 1897–1907, Feb. 2023.

[5] Q. Lin, F. Wang, and J. Xu, "Optimal Task Offloading Scheduling for Energy Efficient D2D Cooperative Computing," *IEEE Communications Letters*, vol. 23, no. 10, pp. 1816–1820, Oct. 2019.

[6] A. Das *et al.*, "Performance Optimization for Edge-Cloud Serverless Platforms via Dynamic Task Placement," in *Proc. IEEE/ACM CCGRID 2020*, May 2020.

[7] C. Cicconetti *et al.*, "Toward Distributed Computing Environments with Serverless Solutions in Edge Systems," *IEEE Communications Magazine*, vol. 58, no. 3, pp. 40–46, Mar. 2020.

[8] A. Tariq *et al.*, "Sequoia: Enabling Quality-of-Service in Serverless Computing," in *Proc. ACM SoCC 2020*, Oct. 2020.

[9] S. Sarkar *et al.*, "Serverless Management of Sensing Systems for Fog Computing Framework," *IEEE Sensors Journal*, vol. 20, no. 3, pp. 1564–1572, Feb. 2020.

[10] H. Wang, D. Niu, and B. Li, "Distributed Machine Learning with a Serverless Architecture," in *Proc. IEEE INFOCOM 2019*, May 2019.

[11] V. Gupta *et al.*, "Utility-based Resource Allocation and Pricing for Serverless Computing," *arXiv:2008.07793*, Aug. 2020.

[12] T. Elgamal *et al.*, "Costless: Optimizing Cost of Serverless Computing through Function Fusion and Placement," in *Proc. IEEE/ACM SEC 2018*, Oct. 2018.

[13] 3GPP TS 23.288, "Architecture Enhancements for 5G System (5GS) to Support Network Data Analytics Services," version 18.7.0, Sep. 2024.