

Edge/Fog Computing Technologies for IoT Infrastructure II

Taehong Kim ¹, Seong-eun Yoo ² and Youngsoo Kim ^{3,*}

¹ School of Information and Communication Engineering, Chungbuk National University, Cheongju 28644, Republic of Korea; taehongkim@cbnu.ac.kr

² School of Artificial Intelligence, Daegu University, Gyeongsan 38453, Republic of Korea; seyoo@daegu.ac.kr

³ Department of Artificial Intelligence, Jeonju University, Jeonju 55069, Republic of Korea

* Correspondence: pineland@jj.ac.kr; Tel.: +82-63-220-2410

The prevalence of smart devices and cloud computing has led to an explosion in the amount of data generated by IoT devices. Moreover, emerging IoT applications, such as augmented and virtual reality (AR/VR), intelligent transportation systems, and smart factories, require ultra-low latency for data communication and processing. Edge/fog computing is a new computing paradigm where fully distributed fog/edge nodes located near end devices provide computing resources. By analyzing, filtering, and processing at local fog/edge nodes instead of transferring tremendous amounts of data to centralized cloud servers, fog/edge computing can significantly reduce processing delay and network traffic. With these advantages, fog/edge computing is expected to be one of the key enabling technologies for building IoT infrastructure.

In order to examine the recent research on edge/fog computing technology for building IoT infrastructure, this Special Issue published seven articles after a thorough review process. The selected articles cover diverse topics such as the distributed orchestrator model, task offloading, task scheduling, and mutual authentication protocol on edge/fog computing infrastructure, which represent the recent trends and the state-of-the-art algorithms of edge/fog computing technologies.

The article “Distributed Agent-Based Orchestrator Model for Fog Computing” by Agnius Liutkevičius et al. [1] presents a distributed agent-based orchestrator model that enables flexible service provisioning in a dynamic fog computing environment, accounting for the resource constraints and security levels of fog nodes. Unlike previous works that rely on a central control entity, the proposed model distributes decision-making agents across every fog node, promoting efficient decision-making, energy conservation, reduced delay, and improved system performance by placing decision-making at the closest fog node. The proposed model is implemented and tested on real hardware as a prototype, demonstrating its superior performance in terms of response latency and computational overhead when compared to the batch algorithm.

The article titled “Dynamic Task Offloading for Cloud-Assisted Vehicular Edge Computing Networks: A Non-Cooperative Game Theoretic Approach” [2] presents an efficient dynamic task offloading scheme based on a non-cooperative game (NGTO) to overcome the challenges of determining a real-time processing location for offloaded tasks in the vehicular edge computing (VEC) environment, where high vehicle mobility and overload problems are prevalent. The authors propose the best response offloading strategy that dynamically adjusts the task-offloading probability between the multi-access edge computing (MEC) server and the cloud to maximize utility for each vehicle. To design the VN (vehicular network) model, they considered the movement of vehicles at different speeds and utilized the vehicle-to-RSU (V2R) communication mode for local road-side units (RSU) computing. Vehicles can offload their computing tasks via a base station (BS) or RSUs. Through EdgeCloudSim simulations, the proposed scheme achieved performance guarantees, reduced response time, and task failure rates.



Citation: Kim, T.; Yoo, S.; Kim, Y. Edge/Fog Computing Technologies for IoT Infrastructure II. *Sensors* **2023**, *23*, 3953. <https://doi.org/10.3390/s23083953>

Received: 23 March 2023

Accepted: 3 April 2023

Published: 13 April 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

The next article, “Collaborative Task Offloading and Service Caching Strategy for Mobile Edge Computing” by Xiang Liu et al. [3], focuses on the collaborative task offloading problem assisted by a dynamical caching strategy in mobile edge computing (MEC) and proposes a two-layer computing strategy called joint task offloading and service caching (JTOSC) to effectively reduce the maximum delay of all users. The outer layer in JTOSC iteratively updates the service caching decisions using Gibbs sampling, while the inner layer adopts a fairness-aware allocation algorithm and an offloading revenue preference-based bilateral matching algorithm to obtain a better computing resource allocation and task offloading scheme. The simulation results indicate that the proposed strategy outperforms existing methods in terms of maximum offloading delay, service cache hit rate, and edge load balance.

The article “Local Scheduling in KubeEdge-Based Edge Computing Environment” by S. Kim and T. Kim [4] investigates the performance of KubeEdge in terms of computational resource distribution and latency between edge nodes. The study reveals that forwarding traffic between edge nodes leads to degraded throughput and increased service delay in an edge computing environment. To mitigate this problem, the authors propose a local scheduling scheme that processes user traffic locally at each edge node. The performance evaluation demonstrates that the local scheduling scheme outperforms the existing load-balancing algorithm in KubeEdge in an edge computing environment.

The following article, “Latency-Aware Task Scheduling for IoT Applications Based on Artificial Intelligence with Partitioning in Small-Scale Fog Computing Environment” [5], proposes a task scheduling algorithm that uses artificial neural networks with partitioning capabilities to enable multiple edge servers to learn and calculate hyperparameters in parallel. This approach is designed to reduce scheduling time and improve service level objectives while minimizing energy consumption in small-scale fog computing environments. The performance evaluations demonstrate that the proposed algorithm outperforms existing scheduling algorithms in terms of latency reduction and energy consumption.

The article “Autonomous Mutual Authentication Protocol in Edge Networks” [6] addresses the challenge of designing a mutual authentication protocol for the large number of autonomous devices in edge networks. To overcome this, the authors propose a novel decentralized protocol based on a public key system, octet-based balanced-tree transitions, challenge–response mechanism, device unique ID, pseudo-random number generator, time stamps, and event-specific session keys. The protocol is analyzed for its effectiveness against typical attacks, and its feasibility and effectiveness are demonstrated through experimental analysis in a real-world scenario with various edge devices.

The next article, “Cooperative Downloading for LEO Satellite Networks: A DRL-Based Approach” [7], addresses the challenge of transmitting collected data to ground stations in low Earth orbit (LEO) satellite-based applications, where high satellite mobility results in insufficient download time. To address this issue, the authors propose a deep reinforcement learning (DRL)-based cooperative downloading scheme that formulates a Markov decision problem (MDP) with the objective of maximizing the downloaded data. They adopt a soft actor critic (SAC)-based DRL algorithm to learn the optimal approach. The evaluation results demonstrate that the proposed scheme achieves a higher utilization of the satellite’s contact time compared with other schemes, making it an effective approach for LEO satellite-based applications.

Funding: This research has been supported by the Research Grant of Jeonju University in 2022 and the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2020R1F1A1048179).

Acknowledgments: The Guest Editors would like to thank all the authors for their submissions to this Special Issue. We are also grateful to the reviewers for their valuable and constructive comments on the manuscripts.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Liutkevičius, A.; Morkevičius, N.; Venčkauskas, A.; Toldinas, J. Distributed Agent-Based Orchestrator Model for Fog Computing. *Sensors* **2022**, *22*, 5894. [[CrossRef](#)] [[PubMed](#)]
2. Hossain, M.D.; Sultana, T.; Hossain, M.A.; Layek, M.A.; Hossain, M.I.; Sone, P.P.; Lee, G.-W.; Huh, E.-N. Dynamic Task Offloading for Cloud-Assisted Vehicular Edge Computing Networks: A Non-Cooperative Game Theoretic Approach. *Sensors* **2022**, *22*, 3678. [[CrossRef](#)] [[PubMed](#)]
3. Liu, X.; Zhao, X.; Liu, G.; Huang, F.; Huang, T.; Wu, Y. Collaborative Task Offloading and Service Caching Strategy for Mobile Edge Computing. *Sensors* **2022**, *22*, 6760. [[CrossRef](#)] [[PubMed](#)]
4. Kim, S.-H.; Kim, T. Local Scheduling in KubeEdge-Based Edge Computing Environment. *Sensors* **2023**, *23*, 1522. [[CrossRef](#)] [[PubMed](#)]
5. Lim, J. Latency-Aware Task Scheduling for IoT Applications Based on Artificial Intelligence with Partitioning in Small-Scale Fog Computing Environments. *Sensors* **2022**, *22*, 7326. [[CrossRef](#)] [[PubMed](#)]
6. Sheu, R.-K.; Pardeshi, M.S.; Chen, L.-C. Autonomous Mutual Authentication Protocol in the Edge Networks. *Sensors* **2022**, *22*, 7632. [[CrossRef](#)] [[PubMed](#)]
7. Choi, H.; Pack, S. Cooperative Downloading for LEO Satellite Networks: A DRL-Based Approach. *Sensors* **2022**, *22*, 6853. [[CrossRef](#)] [[PubMed](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.