



HAL
open science

An AI-Assisted Model for Task Offloading Decision Making in Edge Computing

Messaoud Babaghayou, Nouredine Chaib, Mohamed Amine Ferrag, Leandros Maglaras

► **To cite this version:**

Messaoud Babaghayou, Nouredine Chaib, Mohamed Amine Ferrag, Leandros Maglaras. An AI-Assisted Model for Task Offloading Decision Making in Edge Computing. Colloque sur les Objets et Systèmes Connectés 2023, Institut Supérieur des études technologiques de Sfax; Institut Supérieur des études technologiques de Mahdia, Jun 2023, Mahdia, Tunisie. hal-04219825

HAL Id: hal-04219825

<https://hal.science/hal-04219825v1>

Submitted on 27 Sep 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

An AI-Assisted Model for Task Offloading Decision Making in Edge Computing*

Messaoud Babaghayou^{1,*}, Nouredine Chaib¹, Mohamed Amine Ferrag² and Leandros Maglaras³

(1) LIM Lab., Amar Telidji University, Laghouat 03000, Algeria

(2) Technology Innovation Institute, Masdar City 9639, Abu Dhabi, United Arab Emirates

(3) Centre for Cybersecurity, IoT and Cyber-physical systems (CIC), Edinburgh Napier University, Edinburgh EH10 5DT, UK
 {messaoud.babaghayou & n.chaib}@lagh-univ.dz, mohamed.ferrag@tii.ac, l.maglaras@napier.ac.uk

Abstract - Edge computing (EC) serves as a promising complement to cloud computing, yet the challenge of task offloading decision-making persists. While task offloading can extend device lifespan, it may introduce delays that surpass acceptable application thresholds. To address this challenge, we propose a machine learning (ML) technique that leverages key factors such as network latency, battery level, and device location to inform task offloading decisions. This novel ML technique, which is exclusively presented in this paper, holds potential for enhancing task offloading decision-making. Furthermore, the paper discusses future work, including testing and validating the proposed technique, as well as conducting an investigation into the impact of varying the number of mobile users within an edge-enabled environment.

Index Terms - Edge computing, Task offloading, Resources optimization, ML technique.

I. INTRODUCTION

Cloud computing (EC) has emerged as a prominent and transformative concept in the field of distributed computing. It stands apart from traditional service-oriented and grid computing paradigms, presenting opportunities for synergistic collaborations. While cloud computing offers immense potential, it is not without its share of challenges that demand careful attention [1]. Figure 1 illustrates a potential architecture that can shape the EC paradigm.

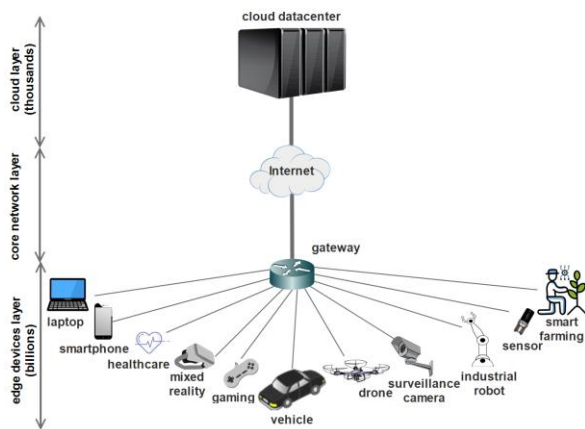


Fig. 1 Cloud computing architecture with regards to trend applications.

Edge computing (EC) stands as a revolutionary technology within the computing domain, offering compelling solutions to address the limitations faced by smart devices. By

offloading computationally intensive tasks to cloud servers, edge computing effectively reduces the total energy [2] consumption and extends the battery life of these devices. This enables seamless operation, even in energy-constrained scenarios. Furthermore, the integration of Mobile Edge Computing (MEC) [3] servers in close proximity to end devices serves to minimize transmission latency, resulting in enhanced real-time responsiveness [4].

EC unlocks a multitude of applications across various domains, such as the Internet of Things (IoT), industrial systems, surveillance, autonomous vehicles [5], XR technologies (Virtual Reality and Augmented Reality), unmanned aerial vehicles (UAVs), and more, as depicted in Figure 2. These diverse utilization cases highlight the versatility and potential of edge computing.

Moreover, edge computing brings about the convergence of energy-efficient computation and wireless power transfer, paving the way for seamless integration of smart devices into autonomous environments. Despite a number of challenges [6] and the impact of resources geo-location emplacement [7], this convergence holds immense promise for the future, propelling the advancement of smart devices and their efficient integration into various autonomous systems [4].

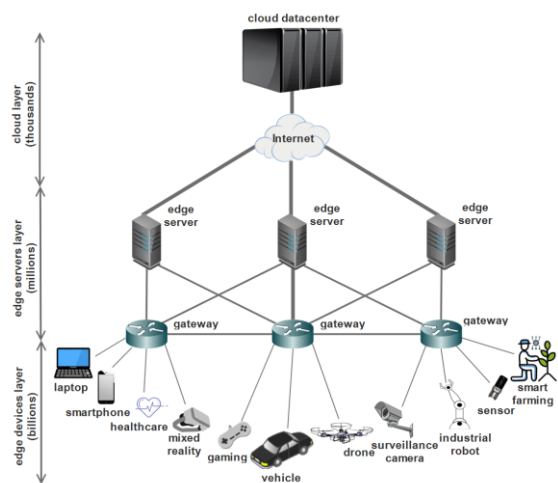


Fig. 2 Edge computing architecture with regards to trend applications.

In dynamic environments with wireless fading channels, making effective decisions between local and remote

* This work is partially supported by NSF Grant #2003168 to H. Simpson and CNSF Grant #9972988 to M. King.

computation becomes a critical task. To adapt to the challenges posed by dense networks and time-varying wireless channel conditions [8], optimal offloading actions are required. Techniques such as reinforcement methods are employed to determine the most suitable offloading strategy. This issue has prompted the research community to explore various solutions, leveraging tools and techniques available in the field of Artificial Intelligence (AI) and its branch of Machine Learning (ML) [9] (with even advanced systems like federated learning [10]). These advanced AI and ML techniques are instrumental in addressing the complexities associated with decision-making in wireless environments.

The contributions of this research can be stated as follows:

- Our proposal revolves around harnessing the power of machine learning (ML) techniques to address the complex task of making offloading decisions in an environment enabled by edge computing.
- We undertake a comprehensive study to examine how changes in the number of mobile users impact an architecture based on the principles of edge computing.
- We outline our forthcoming approach to implementing the proposed technique, which aims to achieve enhanced decision-making capabilities for offloading tasks.

The research paper follows a well-organized structure. In Section II, we provide an overview of existing research conducted in the area of optimizing offloading decisions in edge computing. Following that, we present the system model and introduce a dedicated machine learning (ML) technique specifically designed to tackle the offloading decision challenge. In the subsequent section, we delve into the simulation phase, where we investigate the effects of changing the number of mobile users in the edge computing environment. Lastly, we conclude the study by summarizing the key findings and outlining potential directions for future research.

II. RELATED WORK

The field of edge computing has witnessed remarkable strides, particularly in the realms of task offloading decision-making and resource optimization. Numerous research studies have delved into these domains, delving deep into innovative approaches and techniques. These dedicated endeavors have played a pivotal role in augmenting the efficiency and effectiveness of edge computing systems, enabling them to allocate resources seamlessly and make well-informed decisions when it comes to task offloading. The advancements borne out of these works have laid a solid foundation for improved performance, decreased latency, and heightened resource utilization within edge computing environments. As a result, organizations can now leverage the full potential of edge computing, unlocking unprecedented levels of efficiency and responsiveness. The collective progress achieved through these research efforts continues to drive the evolution of edge computing, opening up new possibilities and shaping the

future of computing paradigms. [11]. We shed light on few of these research works as follows:

To offer a comprehensive understanding of edge computing, the European Telecommunications Standards Institute (ETSI) took a significant step by publishing an introductory technical white paper on Mobile Edge Computing (MEC) in 2014. This authoritative document serves as a valuable resource, providing clear definitions and insights into different facets of MEC. It covers a wide range of topics, including architectures, technologies, platform APIs, interfaces, and even business models associated with the Radio Access Network (RAN) within the context of edge computing. By establishing this foundational knowledge, the white paper plays a crucial role in fostering a deeper comprehension of the intricacies and potential applications of edge computing in real-world scenarios [12].

Alongside edge computing, another influential contender in the field is fog computing, which was first introduced by Cisco in 2012. Fog computing extends the capabilities of the cloud by bringing it closer to the network edge. This paradigm places emphasis on several notable characteristics, such as distributed processing, real-time analytics, seamless integration with the cloud, and efficient network management. By leveraging these features, fog computing empowers organizations to harness the power of computing resources at the network edge, enabling faster data processing, enhanced responsiveness, and seamless collaboration with cloud services [13]. The recognition of fog computing alongside edge computing enriches the technology landscape, providing a broader range of options for diverse computing requirements and dynamic network infrastructures.

Moreover, the relationship between edge computing and 5G is highly symbiotic, with both technologies mutually reinforcing each other. The convergence of edge computing and 5G is of paramount importance, as edge computing plays a vital role in unlocking the full potential of 5G networks. By harnessing the capabilities of edge computing, 5G networks can leverage low-latency and high-bandwidth capabilities, paving the way for a wide range of innovative applications and services, including those in the healthcare domain [14]. The powerful combination of edge computing and 5G holds great potential in revolutionizing industries, elevating user experiences, and spearheading the future of connectivity and digital transformation.

In the nineties, certain studies like [15] and [16] have concentrated on bolstering the performance of mobile applications while conserving energy on mobile devices through collaborative computing with remote servers that are within the reaching range [17]. Their innovative endeavors resulted in the creation of Odyssey, a prototype that served as a tangible demonstration of enabling adaptive execution of mobile applications. This application-aware adaptation empowered the dynamic adjustment of application behavior in response to available resources such as CPU cycles, bandwidth, and battery power. Their pioneering work established the groundwork for optimizing mobile application

performance and energy efficiency through intelligent resource management and adaptive execution techniques.

Task offloading in Mobile Edge Computing (MEC) tackles the constraints faced by IoT devices by incorporating energy harvesting technology and advocating for a hybrid energy supply model. Authors in [18] introduced a dynamic task offloading algorithm named DTOME, which aimed to minimize system cost by optimizing local computing, offloading duration, and edge computing decisions. By taking into account both system cost and queue stability, DTOME effectively made task offloading decisions and achieved lower system cost compared to traditional strategies. These findings underscored the importance of energy efficiency and cost optimization in Mobile Edge Computing (MEC) systems, thereby advancing the development of IoT applications.

Within this section, we introduce a comprehensive system model that serves as the cornerstone of our research study. We meticulously constructed this system model to encompass the essential elements and factors crucial for our investigation, as depicted in Figure 3. By establishing a robust system model, we gain a deeper understanding of the various entities that play pivotal roles in the edge-enabled environment.

III. SYSTEM MODEL

In this section, we elucidate the key entities within the realm of edge computing. By doing so, we offer a comprehensive understanding of their respective roles and the impact they have on edge computing systems. Through concise and informative explanations, readers can grasp the significance of these entities within the broader context of edge computing:

- Mobile edge devices: including smartphones (as shown in Figure 3), tablets, and IoT devices, encompass wireless devices with computational capabilities situated at the network edge. These devices act as endpoints for executing computational tasks and offloading them to edge servers. This arrangement facilitates efficient processing, minimizes latency, and optimizes the overall performance of the system.
- Access points: are wireless networking devices that function as gateways connecting mobile devices to edge servers. These access points play a crucial role in facilitating communication and ensuring connectivity between mobile devices and the edge computing infrastructure. By enabling seamless data transfer, offloading capabilities, and efficient resource utilization, they contribute to the smooth operation of the edge computing ecosystem.
- Routers: are integral networking devices that assume the responsibility of forwarding data packets between diverse networks. Their primary function is to direct traffic and establish efficient communication paths within the mobile edge computing architecture. By effectively managing data flow between mobile devices, access points, and edge servers, routers

ensure smooth and reliable connectivity, enabling seamless operation of the entire system.

- Edge servers: refer to computing nodes strategically positioned at the network edge to offer computational resources and services to mobile devices. Their presence facilitates accelerated data processing, minimized latency, and enhanced response times. By bringing computational capabilities closer to end-users, edge servers significantly improve the performance and efficiency of mobile edge computing applications. This proximity enables seamless and rapid execution of tasks, ultimately enhancing the overall user experience.
- The orchestrator: serves as a central entity within the mobile edge computing infrastructure, responsible for managing and coordinating resource allocation, task distribution, and service provisioning. It plays a pivotal role in optimizing the utilization of resources, facilitating efficient task offloading, and ensuring optimal system performance. By dynamically orchestrating the interactions among mobile devices, edge servers, and other components, the orchestrator enables seamless coordination and synchronization, leading to the efficient operation of the entire mobile edge computing ecosystem.
- Cloud datacenter: refers to a centralized facility designed to accommodate numerous servers and abundant computing resources. It serves as a hub for storage, processing power, and a wide range of services necessary for handling data-intensive tasks. By supporting the offloading of computations from mobile edge devices, cloud datacenters enable scalable and resource-rich computing capabilities for edge applications. These facilities play a critical role in enhancing the performance and scalability of edge computing systems, offering the necessary infrastructure to handle complex computational workloads and efficiently process large volumes of data.
- The machine learning model: is a mathematical representation or algorithm that undergoes training using data to gain the ability to make predictions or decisions. These models can be deployed on edge devices or edge servers, bringing intelligent decision-making, data analysis, and pattern recognition tasks to the edge of the network. By leveraging machine learning at the edge, real-time capabilities are enhanced, reducing the dependency on cloud-based processing. This empowers edge devices and servers to perform sophisticated tasks locally, leading to faster response times, improved privacy, and increased efficiency in handling data-intensive workloads.

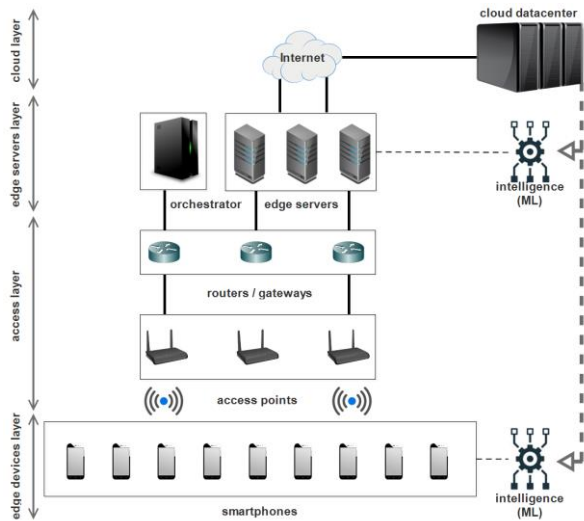


Fig. 3 the taken system model for the study.

IV. METHODOLOGY

In this section, we present our proposed methodology to tackle the task offloading decision problem. Our approach leverages a comprehensive set of features, including application type, task size, edge resource availability, device energy level, and device geo-location, to enhance decision-making. The process, depicted in Figure 4, involves transmitting these features to a cloud datacenter, where a machine learning model is trained and constructed.

The trained model can then be deployed across the edge infrastructure, including both edge devices and edge servers, enabling intelligent and localized decision-making at the network edge. This approach optimizes task offloading efficiency and resource utilization. The training of the machine learning model relies on a carefully curated training set, which plays a crucial role in its effectiveness.

More precisely: we aim at utilizing a specific supervised learning approach known as classification. We construct a machine learning model that is trained using a carefully curated training set consisting of labeled examples. Each example in the training set comprises input data, representing features such as application type, task size, edge resource availability, device energy level, and device geo-location, along with their corresponding desired outputs.

During the training process, the model learns to map the input features to specific classes or categories, allowing it to classify new instances based on their features. By minimizing the difference between the predicted class and the desired class in the training examples, the model adjusts its internal parameters to improve its classification accuracy.

To achieve this, we employ an iterative optimization algorithm, such as gradient descent, which updates the model's parameters based on the errors between predicted and desired outputs. This iterative learning process allows the model to generalize from the training set, capturing patterns and relationships in the data.

By employing this methodology and employing a well-designed training set, we aim to improve the accuracy and

efficiency of task offloading decisions, leading to enhanced edge computing capabilities and better resource management.

It is crucial to acknowledge that this research introduces a novel idea or concept. However, it is important to highlight that the practical implementation and deployment of this idea are slated for the near future. To bridge the gap between concept and reality, we plan to conduct rigorous validation through simulations. This validation process will enable us to test and refine the proposed methodology, bringing us closer to realizing the full potential of this idea.

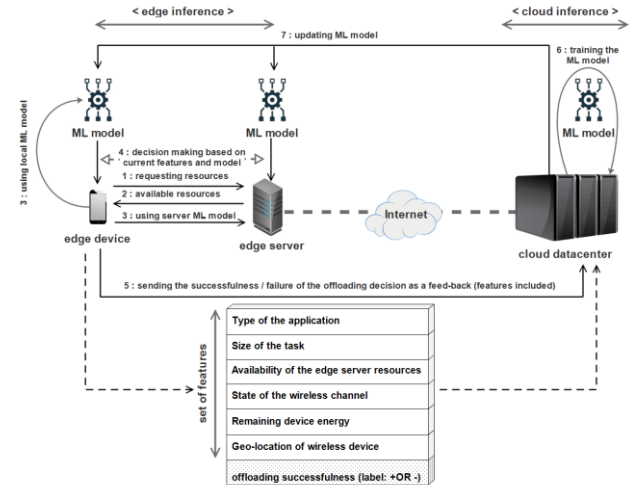


Fig. 4 the methodology of the proposed machine learning technique.

V. SIMULATION

In this section, we undertake a comprehensive analysis of the impact of the number of mobile users on system performance, with a specific focus on two critical metrics: (1) the delay of edge tasks and (2) the received bits per second. By systematically manipulating the number of mobile users, our objective is to gain a deeper understanding of how the system behaves and performs under varying user loads. This investigation will provide valuable insights into the scalability and resilience of the system, offering valuable information for optimizing resource allocation, mitigating delays, and improving overall reliability in dynamic mobile environments. It's important to note that this analysis does not yet incorporate our machine learning technique but serves as a foundation for future enhancements and optimizations.

To achieve this goal, we employ a range of simulation tools, including xFogSim [19] and OMNeTPP [20], to facilitate our research and experimentation. xFogSim, a simulation framework dedicated to fog computing, serves as a powerful platform for modeling and analyzing fog computing environments. It enables researchers to assess various aspects such as resource allocation, task scheduling, energy consumption, and system performance in fog-based systems. Similarly, we utilize OMNeTPP, an open-source and modular simulation framework, known for its versatility in simulating network protocols, communication systems, and distributed systems. With its discrete-event simulation environment, OMNeTPP empowers researchers and developers to

effectively model and analyze complex systems. By leveraging these simulation tools, we can gain valuable insights and evaluate the performance of our proposed methodologies in a controlled and reproducible environment.

Our model, shown in Figure 5 with 60 mobile users, focuses on the impact of changing the number of users in an edge computing simulation. We did not utilize the cloud datacenter capabilities as our main objective is to investigate the effects of mobile user counts on the edge computing system's performance.

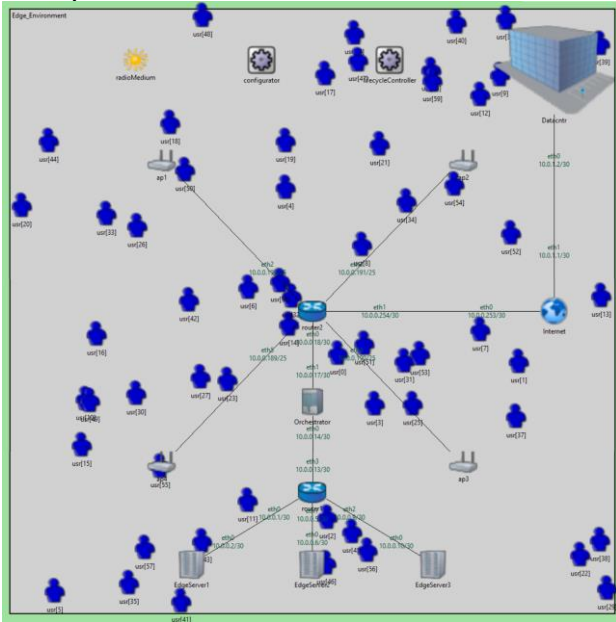


Fig. 5 simulation snapshot of the 60 mobile users instant.

We performed simulations where we varied the number of mobile users from 20 to 100 in two scenarios: one with a task offloading that consumed 1010 MIPS and another with a task offloading that consumed 3010 MIPS. Initially, the task offloading scenarios were set at 1000 MIPS and 3000 MIPS, respectively but we added a fictive 10 MIPS that we assumed is going to be for the training set. In our simulation, ES represents Edge Servers, and their computation capability was fixed at 1000 MIPS each. We also included 4 access points to accommodate the mobile users. The simulation was conducted for 60 seconds in a 500m x 500m area, where the mobility of the users was divided into two types: 20% followed a circular mobility model, while the remaining 80% followed a linear mobility model. The users' departure points were randomly set, and they sent tasks to the orchestrator, which is responsible for dispatching them to the edge servers, at a rate of one task per second.

The results obtained from our experiments are illustrated in Figure 6 for the delay of edge tasks and in Figure 7 for the received bits per second in the 1010 MIPS scenario. The delay of edge tasks exhibits an almost linear increase with the growing number of mobile users for each edge server, ranging from 14 ms to 76 ms. On the other hand, the received bits per second rates for the different edge servers show an ascending trend, starting from 25K and reaching approximately 325K.

Among the edge servers, ES1 consistently receives the highest rates, attributed to the orchestrator's role in task dispatching, followed by ES3, while ES2 records the lowest rate.

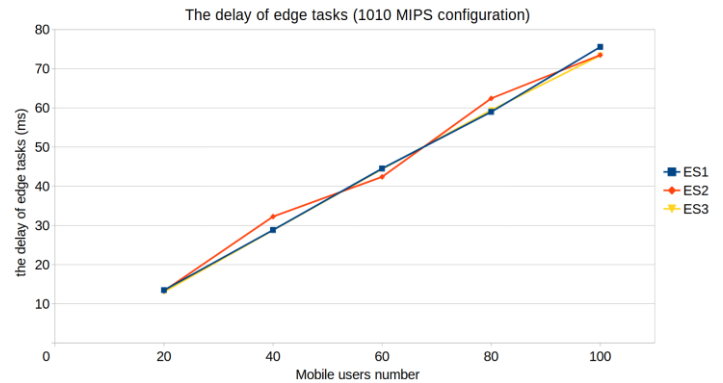


Fig. 6 the achieved delay of edge tasks in a configuration of 1010 MIPS tasks by EdgeServer1, EdgeServer2 and EdgeServer3.

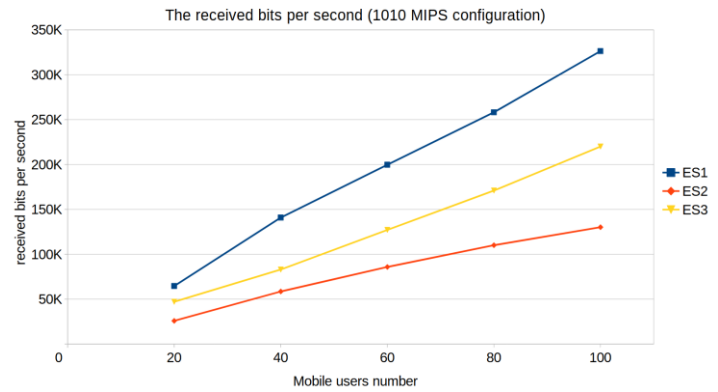


Fig. 7 the achieved received bits per second in a configuration of 1010 MIPS tasks by EdgeServer1, EdgeServer2 and EdgeServer3.

For the 3010 MIPS scenario that is shown in Figure 8 and Figure 9 respectively, it follows the same progression in the delay of edge tasks with slightly higher delay (up to 80 ms) while for the received bits per second rates, it is almost the same (varying from 25K to 325K) with the 1010 MIPS scenario and this is due to the irrelevance between the received rate and the MIPS tasks because the later only has relation with the computation power of the ESs, i.e., causing computation delay.

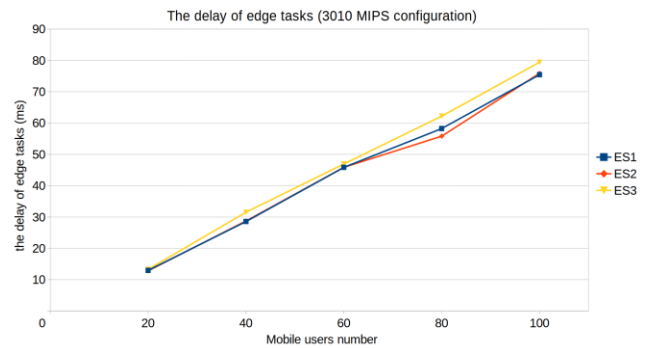


Fig. 8 he achieved delay of edge tasks in a configuration of 3010 MIPS tasks by EdgeServer1, EdgeServer2 and EdgeServer3.

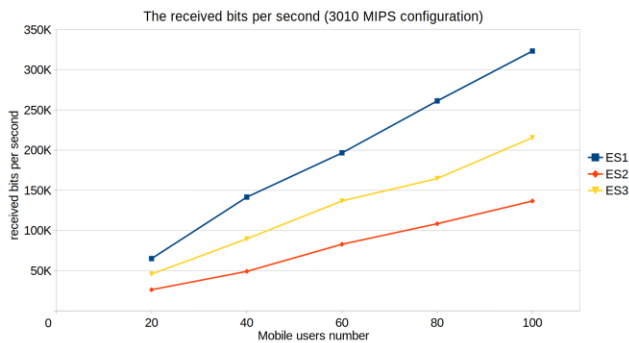


Fig. 9 the achieved received bits per second in a configuration of 3010 MIPS tasks by EdgeServer1, EdgeServer2 and EdgeServer3.

VI. CONCLUSION

To summarize, this research paper introduced an innovative machine learning approach for making task offloading decisions in edge computing environments. The technique took into account crucial factors like network latency, battery level, and device location to optimize offloading decisions and improve device longevity. Through simulations, the study examined the effects of changing the number of mobile users in an edge-enabled environment. The findings provided insights into the relationship between user count and both edge task delay and received bits per second. The investigation was conducted using two scenarios involving tasks with 1010 and 3010 MIPS.

The study found that increasing the number of mobile users resulted in a linear increase in edge task delay for each server. This underscores the need to consider user load when making task offloading decisions to prevent delays beyond acceptable thresholds. Additionally, the varying bits per second rates received by different edge servers demonstrated the effectiveness of the proposed technique in efficiently distributing tasks among edge resources. This research contributes to the field of edge computing by focusing on simulation and investigating the impact of mobile user numbers. However, further testing and validation are required to confirm the effectiveness of the machine learning technique in real-world scenarios.

This research offers valuable insights into task offloading decision-making in edge computing, highlighting challenges and opportunities. The findings lay the groundwork for optimizing resource allocation and enhancing system performance in edge-enabled environments.

REFERENCES

- [1] Tharam Dillon, Chen Wu, and Elizabeth Chang. Cloud computing: issues and challenges. In 2010 24th IEEE international conference on advanced information networking and applications, pages 27–33. Ieee, 2010.
- [2] Ali Al-Shuwaili and Osvaldo Simeone. Energy-efficient resource allocation for mobile edge computing-based augmented reality applications. *IEEE Wireless Communications Letters*, 6(3):398–401, 2017.
- [3] Messaoud Babaghayou, Noureddine Chaib, Nasreddine Lagraa, Mohamed Amine Ferrag, and Leandros Maglaras. A safety-aware location privacy-preserving iov scheme with road congestion-estimation in mobile edge computing. *Sensors*, 23(1):531, 2023.
- [4] Ehaz Mustafa, Junaid Shuja, S Khaliq uz Zaman, Ali Imran Jehangiri, Sadia Din, Faisal Rehman, Saad Mustafa, Tahir Maqsood, and Abdul Nasir

- Khan. Joint wireless power transfer and task offloading in mobile edge computing: a survey. *Cluster Computing*, 25(4):2429–2448, 2022.
- [5] Mahmood A Al-Shareeda and Selvakumar Manickam. A systematic literature review on security of vehicular ad-hoc network (vanet) based on veins framework. *IEEE Access*, 2023.
- [6] Xiaojie Wang, Jiameng Li, Zhaolong Ning, Qingyang Song, Lei Guo, Song Guo, and Mohammad S Obaidat. Wireless powered mobile edge computing networks: A survey. *ACM Computing Surveys*, 2023.
- [7] Messaoud Babaghayou, Nabila Labraoui, Ado Adamou Abba Ari, Mohamed Amine Ferrag, and Leandros Maglaras. The impact of the adversary’s eavesdropping stations on the location privacy level in internet of vehicles. In 2020 5th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM), pages 1–6. IEEE, 2020.
- [8] Ehaz Mustafa, Junaid Shuja, Kashif Bilal, Saad Mustafa, Tahir Maqsood, Faisal Rehman, and Atta ur Rehman Khan. Reinforcement learning for intelligent online computation offloading in wireless powered edge networks. *Cluster Computing*, 26(2):1053–1062, 2023.
- [9] Shuchen Zhou, Waqas Jadoon, and Junaid Shuja. Machine learning-based offloading strategy for lightweight user mobile edge computing tasks. *Complexity*, 2021:1–11, 2021.
- [10] Yinghao Guo, Rui Zhao, Shiwei Lai, Lisheng Fan, Xianfu Lei, and George K Karagiannidis. Distributed machine learning for multiuser mobile edge computing systems. *IEEE Journal of Selected Topics in Signal Processing*, 16(3):460–473, 2022.
- [11] Li Lin, Xiaofei Liao, Hai Jin, and Peng Li. Computation offloading toward edge computing. *Proceedings of the IEEE*, 107(8):1584–1607, 2019.
- [12] Yun Chao Hu, Milan Patel, Dario Sabella, Nurit Sprecher, and Valerie Young. Mobile edge computing—a key technology towards 5g. *ETSI white paper*, 11(11):1–16, 2015.
- [13] Flavio Bonomi, Rodolfo Milito, Jiang Zhu, and Sateesh Addepalli. Fog computing and its role in the internet of things. In *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, pages 13–16, 2012.
- [14] Jianmin Li, Xuecheng Yang, Guangdi Chu, Wei Feng, Xuemei Ding, Xulong Yin, Liangjun Zhang, Wei Lv, Lufei Ma, Liguang Sun, et al. Application of improved robot-assisted laparoscopic telesurgery with 5g technology in urology. *European Urology*, 83(1):41–44, 2023.
- [15] Brian D Noble, Mahadev Satyanarayanan, Dushyanth Narayanan, James Eric Tilton, Jason Flinn, and Kevin R Walker. Agile applicationaware adaptation for mobility. *ACM SIGOPS Operating Systems Review*, 31(5):276–287, 1997.
- [16] Jason Flinn and Mahadev Satyanarayanan. Energy-aware adaptation for mobile applications. *ACM SIGOPS Operating Systems Review*, 33(5):48–63, 1999.
- [17] Messaoud Babaghayou, Nabila Labraoui, Ado Adamou Abba Ari, and Abdelhak Mourad Gueroui. Transmission range changing effects on location privacy-preserving schemes in the internet of vehicles. *International Journal of Strategic Information Technology and Applications (IJSITA)*, 10(4):33–54, 2019.
- [18] Ying Chen, Fengjun Zhao, Yangguang Lu, and Xin Chen. Dynamic task offloading for mobile edge computing with hybrid energy supply. *Tsinghua Science and Technology*, 28(3):421–432, 2022.
- [19] Asad Waqar Malik, Tariq Qayyum, Anis U Rahman, Muazzam A Khan, Osman Khalid, and Samee U Khan. Xfogsim: A distributed fog resource management framework for sustainable iot services. *IEEE Transactions on Sustainable Computing*, 6(4):691–702, 2020.
- [20] Andras Varga and Rudolf Hornig. An overview of the omnet++ simulation environment. In *1st International ICST Conference on Simulation Tools and Techniques for Communications, Networks and Systems*, 2010.