# Dynamic Optimization for Mobile Edge Computing: A Comparative Study

**Khadija Salim Al Yarubi, S. M. Emdad Hossain**
**Department of Information Systems, CEMIS, University of Nizwa, Oman**

**DOI: https://doi.org/10.51584/IJRIAS.2024.90205**
**Received: 24 January 2024; Accepted: 27 January 2024; Published: 29 February 2024**

## ABSTRACT

Edge computing EC is an advanced technology where a server at the edge resource is allocated at the network's edge, closely to end users, mobile devices, sensors, and the developing Internet of Things (IoT). To date, a large number application using edge computing concept has developed and equipped in the market. However, "Edge" refer to continuum network resources as well as computing; through cloud center of data and the sources of data. The Edge is not limited to request service or contents; it is extended to perform tasks like computing from the cloud. One main reason to get well acceptance to the research community is; the edge is able to offload computing, store the data, processing and caching, request distribution and service delivery to the user from the cloud.

In this paper, extensive experiments are conducted which leads to a comparison of several methods in order to solve two problems e.g. Power Allocation problem using Non-Cooperative Game model based on sub-Gradient NCGG and Nertia-Weighted Particle Swarm Optimization PSO methods. Similarly, Joint Request Offloading and Computing Resource Scheduling JRORS using MO-NSGA and Binary article Swarm Optimization BPSO will explored to check their effectiveness and efficiency to our day to day computing operation.

**Keywords:** edge, computing, internet of things, offloading, sub-gradient

## INTRODUCTION

The network-based centralized architecture all data transfers to the data center (cloud), where it makes use of its powerful supercomputer to address the issues with computation and data storage, which permits the services of the cloud to generate financial gains. Traditional cloud computing, however, has a number of drawbacks in the context of (IoT)[1]:

- Latency: High real-time needs are present in novel applications in IoT scenarios. In the classic model of cloud computing, in order to obtain responses, applications transmit information to the datacenter, which increase system latency.
- Bandwidth: The bandwidth of the network will be under a lot of strain when the massive amount of edge devices' data transmitted in real-time manner to the cloud.
- Availability: Obtaining the serves has become a crucial aspect when more and more Internet services are being hosted in the cloud server. Siri is one of the example, when the user will feel frustrated if for short period of time the service is not available. Therefore it is one of the big challenges for the providers of cloud serves to promise the 24/7 availability.
- Energy: As mentioned in the research of Sverdlik, the United States datacenter consumption of energy will increase 4% by 2020 [2], which means that data centers needs lots of energy, and as the transmission and computation increases, the growth of cloud computing will be constrained by energy usage.
- Privacy and Security: The data from thousands of connected end device are directly associated to the life's of the users. The indoor cameras are one of the example, as transmitting the vedios to the clouds

from the houses will raise the possibility of user privacy being compromised. With End Users (EU) General Data Protection Regulation (GDPR) enforcement [3], Privacy and data security concerns are becoming more significant for the cloud service providers.

These five constrains have increased the potentials of the edge computing. Which simply means the data processing goes all the way to the network's edge. Edge computing went through a rapid development sense 2014, in order to minimize the charges of latency and bandwidth, pointing on the limitation of the data center of cloud computing capability, to protect the user privacy as well as increase the availability of the data[1]. This research aim is to find a robust optimization method or mechanism by considering all five state above.

## BACKGROUND

The development of edge computing included phases presentation, definition, and generalization of dormancy. Backtracking to the proposal of Akamai [4] in 1998 of the Content Delivery Network (CDN), it is a caching network based on the internet, it is installed in several locations and directs users to the closest cache server using the load balancing, content distribution, scheduling, and other functional components of the central platform. CDN can enhance user access response time and minimize network slowness.

Ravi et al.[5] initially presented the idea of function caching and used it to create a customized solution for managing the mailbox to reduce the time of data processing as well as the bandwidth. satyanarayana et al.[6] introduced the idea of Cloudlets, as a dependable and resource-rich host, positioned at the network's edge, linked to the internet, and usable from a mobile device to provide services. It is known also as "Small Cloud" that can assist the users with the service similarly as the server of the cloud do, at this point the edge computing emphasized the downstream (the functions are served from the server of the cloud to the end device) to minimize the bandwidth and time.

As mentioned in [7], Edge computing's premise is that computation should be near to the source of the data , with the word "edge" referring to any resource of the network that have been computed all the route between the data source and the cloud server. In this situation, information from the senses near the data source are transformed from unprocessed signals to information with context. Edge data then brought about a rapid rise in the context of IoT.

The way of increasing the processing capability of the data closer to the data sources is what the researchers have been exploring, to find the challenges in the offloading computing and transmission of data. The illustrated models are mobile edge computing (MEC), Cloud Computing, and fog computing. (MEC) is a topology of network that offers cloud computing and information services close to mobile users in the network with wireless access [8]. According to its place within a network and closeness to mobile consumers, it can increase service quality and user experience by reducing latency and achieving greater bandwidth. The construction of edge servers—whose architecture and hierarchical structure are similar to those of an edge computing server—along the way from the cloud server to the edge device for the computation has been pointed out by the MEC Consequently, it was viewed as an essential part of edge computing. Fog Computing defined by Cisco in 2012 as a fully platform for virtualized computing for lowering the amount of communication between cloud computing centers and mobile users in order to move tasks from cloud computing centers to network edge devices, it reduces the demand on required networks' bandwidth and energy consumption, The two types of computing—fog computing and edge computing—share many characteristics but Fog computing is concern more on improving the data transfer, on the other hand Edge computing take into consideration the network demands and computation needs of both infrastructure and end devices, including end device, edge, and cloud cooperation [9]. On the same period, The Chinese Academy of Sciences launched a ten-year project for strategic priority research called

Next Generation Information and Communication Technology (NICT) initiative. Its primary goal is to conduct research for the project’s “cloud-sea” Computing System. By collaborating with and integrating cloud computing technologies and the Sea Computing system, it seeks to advance cloud computing. “Sea” refers to client side enhancements that have human and physical world facing components. While Edge Computing concentrates on the data channel between “Sea” and “Cloud,” it focuses on the two ends, “sea” and “cloud” [10].

In 2013, from the Pacific Northwest National Laboratory Ryan La Mothe in a two page internal report, proposed the phrase “Edge-Computing.”, Where the initially modern Edge computing was developed. On this period, for Edge computing two aspects where introduced ,upstream of IoT and downstream of Cloud serves [11].

Since 2015, Edge computing is experiencing tremendous growth and is receiving significant attention from both academia and business [1]. In August 2016, Intel and the National Science Foundation (NSF) collaborated on a wireless edge network an information center network (ICN-WEN) [12]. Two months after, Three issues were the main emphasis of the Grand Challenges in Edge Computing workshop conducted by NSF: What edge computing will look like in five to ten years, the major obstacles to achieving the goal and the most effective strategies for attracting those obstacles in a cooperative manners[13]. On the same period, First ACM/IEEE Symposium on Edge Computing (SCE) was organized in collaboration between ACM and IEEE, Since then, A track or workshop on edge computing has been added to the major conferences of ICDGS, INFOCOM, Middle Ware, and other significant international conferences. In January 2018, Science Press released the first textbook on edge computing ever [1].

In 2019, After the Open Fog Consortium was established in 2015 by Cisco, ARM, Dell, Intel, Microsoft, and Princeton University, it was combined into the Industrial Internet of Things (IIoT) [3].

In March 2019, The Cloud Native Computing Foundation (CNCF) sandbox has accepted a Kubernets native edge computing framework [14]. Edge computing is significant for the health area as well, as demonstrated by the addition of the Edge track to the Bio-World conference and Expos in the same year [15].

Outside the factory facility, to satisfy the requirements for network integration between fixed and mobile, [16] suggested an architecture for an out-of-factory 5G edge network based on Edge Computing (EC) and 5G LAN technologies; Inside the factory facility, to fulfill the network needs for the production scenario, they offered an Edge computing-based in-factory 5G edge network architecture that is highly reliable, Ultra Reliable Low Latency Communication (URLLC) and Time Sensitive Network (TSN) technologies are deterministic and low latency; Then they introduced a comprehensive approach to factory manufacturing. Moreover, a real-world example of how a gantry crane could be controlled remotely was presented. It was discovered that the introduction of 5G MEC significantly reduced the delay of the remote end-2-end operation of the gantry crane, additionally, production efficiency increased. Meanwhile, Remote control took the place of artificial aerial work, enhancing industrial safety and lowering labor costs.

[17] showed a real-world deployment scenario example and some gathered statistics. Edge computing has the potential to help with green networking and communication by minimizing the amount of network resources required if the services were constantly housed in the cloud. They concentrated on resource efficiency for platforms using green edge computing through autonomous lifecycle management, which is a crucial component of edge computing. They provide a ground-breaking architecture and solution that specifies a perfect configuration for every type of workload in order to ensure resource efficiency and the achievement of Service Level Objectives (SLO)s. based on MEC infrastructure, [18] have looked into the issue of Virtual machine Replica Copies (VRCs) deployment in edge networks. In order to reduce overall data traffic, they suggest using an online deep reinforcement learning placement (DRLP) method to deploy VRCs on edge servers. The proposed online deep reinforcement learning method outperforms the existing

algorithm in terms of computation time and transmission delay, according to extensive experimental results. It can achieve a nearly optimal solution in a shorter amount of time than the enumeration method.

In [19], the authors presented a framework for creating a sophisticated and complete mobile edge computing system that makes use of 5G networks to enable real-time energy-efficient, offloading of safe computations. The outcomes of the experiments support the proposed work's significant accomplishments. Energy efficiency and the Wifi Demonstrate outperformed long-term evolution (LTE). Lowered energy efficiency numbers represent more energy-conscious operation, which optimizes resource consumption and increases battery life.

With the progressive maturation of AI technology and the ongoing increase of edge device processing capacity, MEC will continue to play a significant role in more new scenarios and will be more widely and successfully deployed in diverse industries [20] which is also indicating the need of dynamic optimization along with its robustness.

## METHODOLOGY

Due to the widespread use of wireless communication technology and the abundance of sensor options, mobile devices in the Internet of Things (IoT), including smartphones, smart automobiles, and unmanned aerial vehicles, can connect to the Internet using low-power wide area networks or cellular networks. There are now more and more mobile application types, which has led to strict requirements for processing speed and real-time computing. Because of the strict specifications for minimal latency and excellent accuracy, offloading the processing to the cloud is difficult. Consequently, mobile edge computing, or MEC, is an exciting new technology that can successfully address the drawbacks of conventional mobile cloud computing. At the network's edge, cloud service suppliers and mobile network operators collaboratively supply strong computational and communication abilities, such as Base Stations (BSs) and local wireless Access Points (APs). Through high-speed wireless access networks, mobile phones can obtain the services and resources for computing in practically adjacent to the network edge . Therefore, ultralow latency and flexible processing for mobile device queries requiring a lot of computation are brought about by leveraging edge computing to offload computation requests [21].

We examine a Ultra Dense Edge Computing (UDEC) network, as seen in Fig. 2, which is made up of a group of phone users, U; a collection of micro-BSs (referred to as BSs in the future), N; and a macro-BS, C, that has a deep cloud. Every BS is thought to cover a specific place, and a user of mobiles should only be connected to one zone.

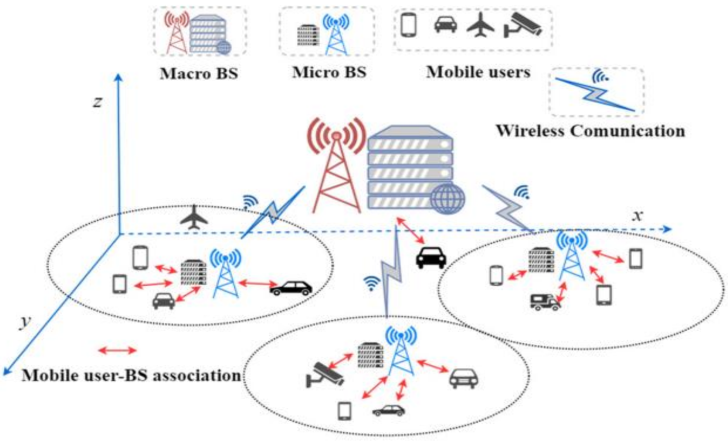


Fig.1 The model of the proposed system

The server in the edge can be either a physical or virtual computer with computational capabilities. It is assumed that the transport links connect its related BS, enabling a nonlocal BS to serve a mobile user. Every mobile user has the option to offload computational requests to a nearby BS. As in [22] Assuming that the macro-BS serves serving as the main manager, it is in the position of gathering task data, computing edge cloud resource information within BSs, and monitoring the network's performance. In particular, U = {1, 2,..u} and N = {1, 2,…, n}, respectively, represent the collection of mobile users and BSs. We suppose that, given qu = <wq,sq, prq, Tgq, Tbq>, every user u ∈ U generates one computing request at a time. Where 1) wq refer to the q request workload: the computation needed to fulfill the request. 2) sq refer to the size of input data. 3) prq refer to the importance of the multiple requests : the priority. 4)Tgq are Optimal delay thresholds and Tbq are acceptable delay thresholds. Taking into account the fact that mobile users' positions change frequently, We utilize pt u = (xu, yu, 0) to indicate where mobile user u is at t period of time. Every BS is in fixed place of BSn shown as, pt n = (xn, yn, H) with the same aspect of h.

However, to implement the method MATLAB R2020b is used, with optimization toolbox and Global optimization toolbox, then building NCGG, MO-NSGA , PSO and BPSO algorithms to assess the efficacy of the suggested techniques. The simulations supposed to run on 16 GB of RAM machine. It is also taken into consideration a multi-zone computing edge system with a macro-BS, several BSs, and users of mobiles. Every BS has an edge node and serves a certain area. Based on the user's location and the area served by the BS, the users were quantized inside a zone linked to the BS. Finally NCGG, MO-NSGA, PSO, and BPSO algorithms applied in MATLAB environment in four different approaches.

1. Simulated Annealing Power Allocation (SAPA): The Noise Power problem can be effectively solved by the conventional heuristic approach known as simulated annealing (SA). The Power Allocation problem, which is described in the Algorithm Efficiency Section -A, is solved by applying SA.
2. Yalmip: Lofberg created Yalmip, a free optimization tool that can handle multiple-objective optimization issues. We solve the JRORS problem, as stated in Algorithm Efficiency Section-B, using Yalmip.
3. The Greedy RS Strategy (ROGS) and Random RO [26]: In order to promote system welfare, Mobile requests from users are arbitrarily distributed among the BSsand every BS carefully plans the use of its computational resources to enhance system utility.
4. Bisection-Based RS Strategy (HOBS) and Heuristic RO [26]: The local optimum can be found in polynomial time using a new heuristic technique called RO, and the bisection approach can be used to solve RS.

## EXPERIMENT DESIGN AND RESULTS

Two (2) different set of experiments have conducted within the environment explained above. The experiments are NCGG & PSO Performance for PA and MO-NSGA & BPSO Performance of JRORS.

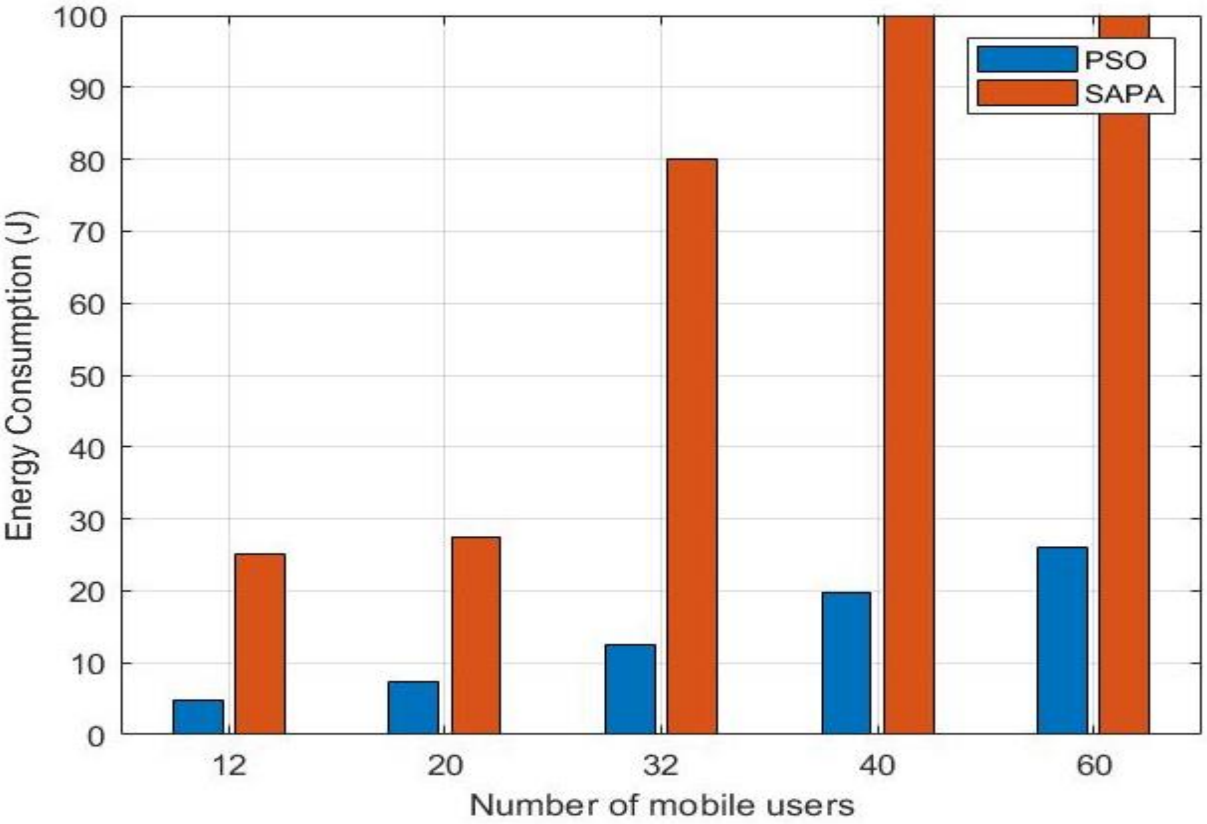
**NCGG & PSO Performance for PA:** The maximum power (pmax) of device users is 5 W in this instance. The efficacy of the suggested NCGG in comparison to the SAPA is displayed in. It goes without saying that as the number of users rises, so does the overall transmission energy consumption. decreases, the overall amount of energy used for transmission rises. Additionally, it is noted that under varying mobile user counts, NCGG's energy consumption is consistently lower than SAPA's, suggesting that NCGG can provide a superior PA outcome in terms of energy savings than SAPA. In NCGG, every mobile user finds its ideal transmitting power and reaches the Nash equilibrium by progressively decreasing from pmax to the subgradient direction. In SAPA, on the other hand, the stable solution that results from random optimization might be a local optimal solution. In the event that no special statement is present, the power policy P* that NCGG obtained serves as the basis for the ensuing trials.

In another experiment with same pmax (5W). The performance of the suggested Weighted Inertia PSO in comparison to the SAPA, when comparing the number of devices users, it is found that PSO's energy consumption is consistently lower than SAPA's. This suggests that PSO outperforms SAPA in terms of energy savings.

**MO-NSGA & BPSO Performance of JRORS:** For this instance, all mobile users offload the same profile request with wq = 1500 (Magacycles), Iq = 700 (KB), Tgq = 0.5 (s), and Tbq = 0.65 (s), and all BSs have the same computing capacity, i.e., Rn = 70 GHz. The ratio of total requests within the request's tolerated delay to the number of finished calculations is how the response rate is defined. Examine MO-NSGA's performance, including system throughput and reaction time, in relation to the remaining three methods using varying numbers of mobile users. Noticed that Yalmip's performance as an optimization tool is limited. This is due to the nonconvex continuous relaxation in the JRORS issue, which implies that there is no guarantee that the Yalmip branching process will discover a globally optimal solution. Furthermore, both HOBS and MO-NSGA outperform ROGS and Yalmip in terms of performance. It is evident that as the number of users rises, so does the welfare of the system. Notably, MO-NSGA can attain a high response rate even when there are a lot of mobile users, which further illustrates the extensibility of MO-NSGA.

Similar experiment conducted to check the effect of number of mobile user where the same profile request is offloaded by all mobile users with wq = 1500 (Magacycles), Iq = 700 (KB), Tgq = 0.5 (s), and Tbq = 0.65 (s), and all BSs have the same processing capacity, that is, Rn = 70 GHz. The ratio of finished calculations to all requests made within the request's acceptable latency is known as the response rate. The performance of Binary PSO, including response rate and system condition. It is evident that as the number of users rises, so does the welfare of the system.

However, after such complex and detailed experiments it is clear that when the energy consumption is increasing with the increase of the mobile users. Comparing SAPA to NCGG, NCGG energy consumption is slightly lower than the SAPA in with the 5W power use, Thus the difference between the SAPA and PSO is notably huge, where PSO show very low power consumption comparing to SAPA, as shown below:



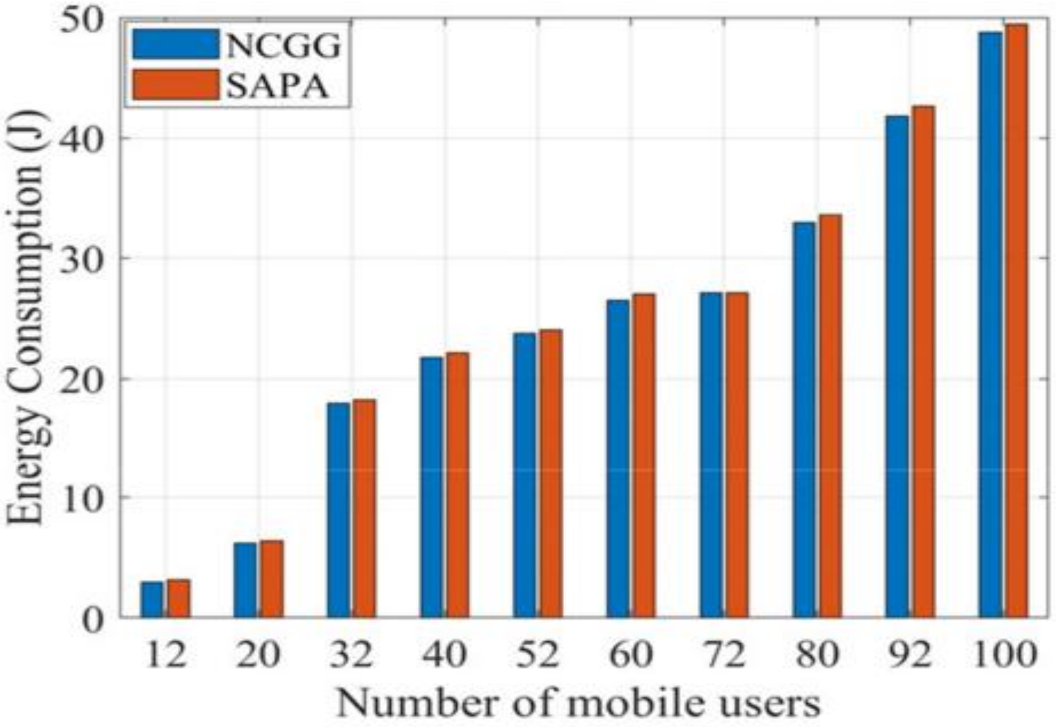
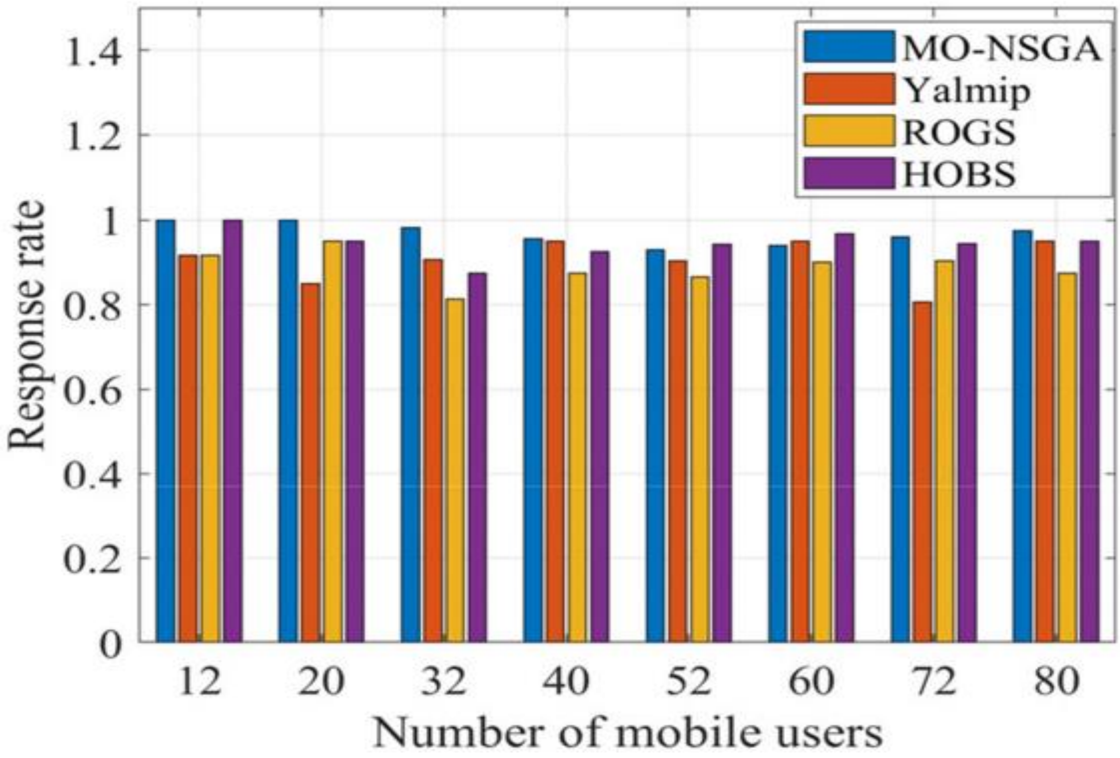


Fig.2 Number of mobile users and energy consumption

Further, the figure below shows the different approaches affects the response rate. With BPSO the response rate is decreasing as the number of the mobile users is increasing, the other hand, the other approaches vary in the response rate by the increasing of the users.



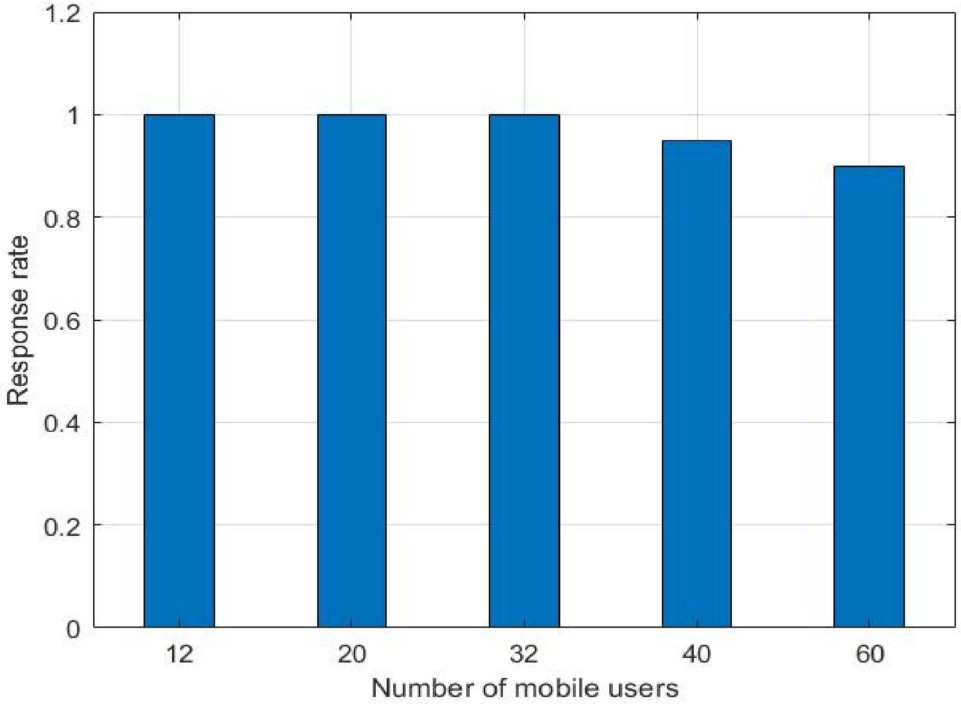


Fig 3 Number of mobile users and response rate

## CONCLUSION

After a set of consecutive experiments under the 5G architecture, we examined at a UDEC network with a macro-BS, numerous micro-BSs, and a sizable user base of mobile users. Specifically, we studied the interference that occurs between users of mobiles and BSs while using the protocol of NOMA. To address this, in the beginning PA problem was created and then an NCGG solution was offered. Further, by creating a mixed-integer nonlinear program, a problem of simultaneously optimizing the RO for mobile users and the computing RS at the micro-BSs was developed. Comparing SAPA to NCGG, NCGG energy consumption is slightly lower than the SAPA in with the 5W power use. With BPSO the response rate is decreasing as the users of mobile number is increasing, on the other hand, the other algorithms vary in the response rate by the increasing of the users. At this stage it is obvious to say that the outcome of this research will lead to a new era especially for dynamic optimization in mobile edge computing. Our future research aim on this topic will be in more focused towards the accuracy of the identified methods. Similarly, to make it published for research community in the area of machine learning, data science, edge comporting. Successful updating and establishment of this approach will have enormous contribution towards the advancement of overall computer science research which in fact; may work for research community in multiple sectors/areas.

## REFERENCES

1. Malarya, A., Ragunathan, K., Kamaraj, M. B., & Vijayarajan, V. (2021, August). Emerging trends demand forecast using dynamic time warping. In 2021 IEEE 22nd International Conference on Information Reuse and Integration for Data Science (IRI)(pp. 402-407). IEEE.
2. Duggal, N. (2023) Top 18 new trends in Technology for 2023: Simplilearn, com. Available at: https://www.simplilearn.com/top-technology-trends-and-jobs-article (Accessed: 02 October 2023).
3. Shi, W., Pallis, G., & Xu, Z. (2019). Edge computing [scanning the issue]. Proceedings of the IEEE, 107(8), 1474-1481.
4. Zwolenski, M., & Weatherill, L. (2014). The digital universe: Rich data and the increasing value of the internet of things. Journal of Telecommunications and the Digital Economy, 2(3), 47-1.

5. Sverdlik, Y. (2016). Here's how much energy all us data centers consume, Data Center Knowledge
6. Regulation, G. D. P. (2018). General data protection regulation (GDPR). Intersoft Consulting, Accessed in October, 24(1)
7. Vakali, A., & Pallis, G. (2003). Content delivery networks: Status and trends. IEEE Internet Computing, 7(6), 68-74.
8. Ravi, J., Shi, W., & Xu, C. Z. (2005). Personalized email management at network edges. IEEE Internet Computing, 9(2), 54-60.
9. Satyanarayanan, M., Bahl, P., Caceres, R., & Davies, N. (2009). The case for vm-based cloudlets in mobile computing. IEEE pervasive Computing, 8(4), 14-23.
10. Symeonides, M., Trihinas, D., Georgiou, Z., Pallis, G., & Dikaiakos, M. (2019, June). Query-driven descriptive analytics for IoT and edge computing. In 2019 IEEE International Conference on Cloud Engineering (IC2E)(pp. 1-11). IEEE.
11. Hu, Y. C., Patel, M., Sabella, D., Sprecher, N., & Young, V. (2015). Mobile edge computing—A key technology towards 5G. ETSI white paper, 11(11), 1-16.
12. Bonomi, F., Milito, R., Zhu, J., & Addepalli, S. (2012, August). Fog computing and its role in the internet of things. In Proceedings of the first edition of the MCC workshop on Mobile cloud computing(pp. 13-16).
13. Xu, Z. W. (2014). Cloud-sea computing systems: Towards thousand-fold improvement in performance per watt for the coming zettabyte era. Journal of Computer Science and Technology, 29(2), 177-181.
14. [LaMothe, R. (2013). Edge computing.Pacific Northwest National Laboratory.[Online]. Available: http://vis. pnnl. gov/pdf/fliers/EdgeComputing. pdf [Retrieved: March 2014].
15. NSF/Intel Partnership on Information-centric networking in wireless … (n.d.). https://www.nsf.gov/pubs/2016/nsf16586/nsf16586.pdf
16. Chiang, M., & Shi, W. (2016, February). NSF workshop report on grand challenges in edge computing. In Rep.
17. Kumar, S., & Du, J. K. a Kubernetes Native Edge Computing Framework
18. Bio-ITworld Edge Track 2019 , Available online at : https://www.bio-itworldexpo.com/edge#
19. Li, Y., Wang, D., Sun, T., Duan, X., & Lu, L. (2020, October). Solutions for variant manufacturing factory scenarios based on 5G edge features. In 2020 IEEE International Conference on Edge Computing (EDGE)(pp. 54-58). IEEE.
20. Guim, F., Metsch, T., Moustafa, H., Verrall, T., Carrera, D., Cadenelli, N., … & Prats, R. G. (2021). Autonomous lifecycle management for resource-efficient workload orchestration for green edge computing. IEEE Transactions on Green Communications and Networking, 6(1), 571-582.
21. Wu, Y., Zhang, S., Shen, G., & Chen, G. (2022, June). Deep reinforcement learning for online vrc deployment in mobile edge computing. In 2022 IEEE 23rd International conference on high performance switching and routing (HPSR)(pp. 271-276). IEEE.
22. Arun, V., & Azhagiri, M. (2023, July). Design of Long-Term Evolution Based Mobile Edge Computing Systems to Improve 5G Systems. In 2023 2nd International Conference on Edge Computing and Applications (ICECAA)(pp. 160-165). IEEE.