A Study on Efficient Approaches for Distributing Workloads Effectively in Edge Computing Systems

Kavya Lingutla, Vennela Priya Penumuchu, Hima Varsha Nagisetty, Niharika Nunna, and S. R. Reeja(✉)

VIT-AP University, Amaravati, India
{kavya.20bce7650,priya.20bce7645,himavarsha.20bce7279, niharika.20bcd7140,reeja.sr}@vitap.ac.in

Abstract. For the benefits of cloud computing, many enterprise companies have moved their services and apps to the cloud. The centralized cloud architecture experiences high workload, congestion, and delay bottlenecks resulting in high amounts of data and rapidly growing digits of connected devices that consume cloud services. Edge Computing (EC) is consequently presented as a new paradigm to increase cloud capabilities close to the end devices. Here, the task allocation is mentioned as the workload distribution amid innumerable nodes in an edge computing network. Major difficulties in workload distribution include locating each task optimally based on its needs for computing capacity, storing data, and bandwidth of the network, and adjusting to network's continuously changing nature. Algorithms for workload allocation can be centralized, decentralized, hybrid, or based on machine learning. The selection of technique relies on the particular application's pre-requisites. Each approach has advantages and disadvantages. In greater detail, the choice of the best work distribution techniques depends on the configuration and architecture of the edge computing system, namely MEC, joint computing of edge, fog and cloud, P2P EC and much more. As a result, allotting the tasks in edge computing is an intricate, varied, as well as a difficult challenge which calls for delicate balancing act amidst multiple potentially competing goals, inclusive of resource-aware, energy efficiency, machine learning with latency, safety and quality of Experience (QoE). Recent years have seen a rise in the amount of research studies on edge devices' work allocation optimization and performance evaluation. This paper compares and contrasts several methods for work load distribution, algorithms which are much optimized, and the communication network types which are often employed in edge computing systems.

Keywords: Artificial Intelligence-Based Task Allocation · Resource Aware · Energy and Delay Reduction

X. Cheng (Ed): BROADNETS 2024, LNICST 601, pp. 143–156, 2025.
https://doi.org/10.1007/978-3-031-81168-5_14

1 Introduction

A new paradigm known as edge computing (EC) appeared after cloud technology had matured. The availability of processing power as well as the high-bandwidth, low-latency communication links among edge nodes place it in the centre of network research's focus. According to Gartner, there will be 20 times as many smart devices on networks' edges by 2023, and by 2025, 75% of the data produced by businesses will be stored outside of traditional data centres and the cloud. The term "edge computing" mentions a decentralized computational model which enables data processing to take place near the storage alternative to depending on centralized data centres. Aforementioned method is especially beneficial to edge-of-the-network augmented reality applications, Internet of Things systems, and autonomous vehicles that require high bandwidth and low latency. The Edge-Fog-Cloud (EFC) architecture which is shown in Fig. 1, is a distributed computing platform created to enhance the effectiveness IoT, 5G, and other latency-sensitive applications. Edge computing, fog computing, and cloud computing are the three main layers that make up the multi-tiered strategy that is used to achieve this. Each layer has a specific function and manages various parts of computing and data processing.

The workload distribution practices used by edge devices today are examined in this article. To determine which important strategies would be most beneficial for future study and effectiveness of every strategy, we examine the major approaches presented in the literature review. We review and assess the body of recent research on the distribution of workload on edge computing devices. The article offers complete analysis including primary approaches and edge computing techniques for effectively handling challenging workloads.

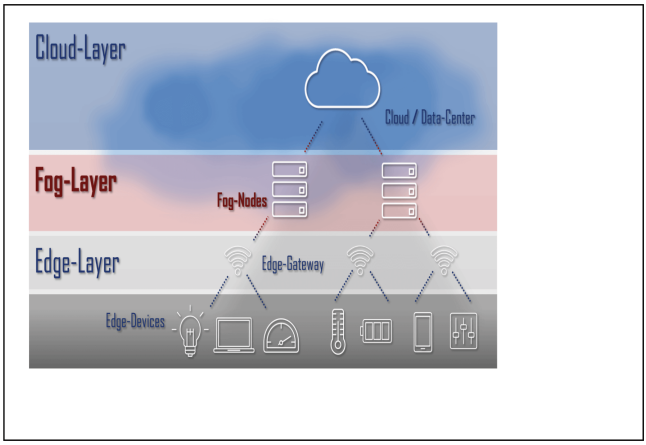

Fig. 1. An edge-fog-cloud computing system's Architecture.

This paper's primary contribution is an overview of contemporary task distribution techniques. This paper specifically stands out on the popular workload distribution techniques and algorithms which are much optimized along with the communication network types incorporated in various edge computing systems, and it only draws on research that

has been published within the last three years. In order to improve the present perception of decision-making while coming to selecting the best workload distribution technique, optimized algorithms, and communication network types based on situation, the paper demonstrates the workload distribution as well as all algorithms that are optimized and techniques which have been recently, most frequently used.

2 Literature Review

The categorization of the job allocation techniques that are used in the latest works and literatures are presented in this section. Each job allocation technique, when combined with the kind of computer network and the deep learning compression techniques, have certain advantages and are used to achieve various goals, including minimizing processing power, energy use, and network delay. The following methods are used to categorize task allocation techniques:

AI/ML: Used some of the algorithms of machine learning for forecasting resource usage and assign work appropriately.

Resource-Aware: distributing tasks is done according to the capabilities of edge devices, such as their battery life, processing speed, and memory capacity.

Distributed: A framework of edge devices are used for distributing jobs in order to improve performance and reduce latency.

Energy-Efficient: Dividing up the work so that edge devices use the least amount of energy possible.

QoS: Task distribution depending on the various tasks needed QoS levels.

An overview of all the workload distribution techniques from all the considered papers is given in Table 1.

2.1 Machine-Learning/Artificial Intelligence-Based Task Allocation

Edge computing with AI and ML enables devices to make choices in real time, minimize data transmission to the cloud, improve security, and offer individualized experiences. This technology ensures the best performance in decentralized contexts for applications like autonomous cars, predictive maintenance, and effective resource allocation. The goal of this research study [9] Using a (DRL) method based on the DQN algorithm, scheduling of the various tasks in edge computing has been optimized. This strategy's main goals are to balance workload distribution, shorten service times, and lower the proportion of tasks that fail to complete. DQN succeeds in attaining these objectives, having been selected for its capability to manage the complex and high-dimensional nature of workload scheduling problems. It is noteworthy that it functions without the requirement for a mathematical model of the environment, instead relying on learnt information from previous actions. DQN outperforms both PPO and DDPG in terms of performance, having the shortest average service time of 0.7 s. Even though the number of devices have been growing, DQN continuously outperforms its competitors with an average VM utilization rate of about 24%. The most promising option for workload scheduling in edge computing scenarios, especially as the complexity and size of the

environment increase, is DQN due to its extraordinary resilience in minimizing the failed task rate, achieving an incredibly low rate of only 3%.

The goal of the research study [10] is to apply a Coalitional Game-Based Service Migration (CGSM) approach that takes into account mobile user reallocation in crowded environments to address the problem of workload balancing in edge computing. The program uses a modified k-means clustering technique to group MEC servers into alliances, locate hotspots, and cooperatively schedule services. In order to improve utilization and load equality across coalition members, it also migrates services to suitable edge servers. Performance analysis shows that, in comparison to other methods, CGSM greatly lowers the number of service migrations and significantly increases user fairness. This method makes use of MEC to process data at mobile edge network and shows future potential for integrating device-to-device technology with service migration for handling large-scale user equipment requirements in crowded circumstances. Research's [13] main goal is to reduce the net cost spent by each individual edge server while offering edge computing services. A brand-new cooperative queueing game strategy is put forth to accomplish this goal. With the help of this strategy, each server's queueing game will have multiple dimensions. These tactics include how much labor is to be done, how much money peers will accept, and how fast computers will be used. The ultimate goal is to determine the judgement of a durable, ideal task distribution for every server on the EC system, and to obtain balance in contexts of costing approaches as well as regulate processing rate to reduce overall performance of the system.

In this study [15], we suggest a container scheduling system and an automatic parameter alteration method to boost serverless computing effectiveness in edge contexts. It introduces the Skippy algorithm, a complex method for placing serverless edge functions for maximum efficiency. It utilizes various kind of priority functions, metadata to allot resources precisely. It illustrates Skippy's usefulness in raising the effectiveness of edge computing deployments through performance comparisons and thorough experimentation. The research highlights the necessity of improving serverless function placement methodologies and the possibility for improvements in dynamic workload distribution as future directions for enhancing performance in edge computing. In this study [18], the positioning of application modules in the edge network is optimized using the BMOPSO algorithm. Results from experiment, which makes use of the Fog simulator, show that BMOPSO performs better than Edge wards and Popularity-based algorithms in matter of implementation time and placement time. The main objective of project was to improve workload placement using swarm intelligence, consequently improving processing times in a variety of areas, such as computation, data propagation, and service deployment in IoT and Edge computing. To sum up, the suggested BMOPSO strategy for workload placement in Edge computing offers a beneficial trade-off across the cloud, edge, and IoT layers. BMOPSO is a great tool for increasing resource utilization and reducing resource waste since experimental results show its effectiveness in maximizing task placement.

This study [1] discusses service pricing, task distribution, and incentive design for the practical use of CVEC in order to keep up with the computing needs of vehicular networks, CVEC integrates MEC into parked automobiles. This results in a new computing paradigm. In CVEC, MEC server is being deployed by an offloading service

supplier and plans parked vehicles as required to manage offloading activities. In order to achieve the best service pricing and workload distribution, the study employs ideal contract layout in light of prospect theory. Comparison of performance shows that, for the lowest PV type, the solution based on Prospect Theory (PT) is almost 90% of the solution based on Expected Utility Theory (EUT). The scope for the future suggests using centralized/distributed machine learning, deep reinforcement learning, and other methods to further optimize. Dividing up the workload. In order to allocate tasks in the most effective way, the study [5] proposes a hierarchical control system to address load balancing issues in edge computing environments using evolutionary algorithms, hidden Markov models, and game theory. This paradigm optimizes task distribution by taking important aspects like service quality, resource efficiency, cost, and energy use into account. A comparison shows this technique is more effective in regards of make span, performance, and cost-effectiveness. In conclusion, this research gives an extensive and priceless answer to the complex work allocation problems in edge computing, promising improved efficiency, cost savings, and service quality, coinciding with the growing significance of edge computing in contemporary IT ecosystems.

2.2 Distributed Based Task Allocation

Caroline Rublein et al. [20] suggested a task allocation approach which is distributed where server works alone and doesn't interact with one another to decide on allocation. Clustering and a two-round bidding technique were suggested. Users post job requests with the necessary resources, a deadline, and utility. To increase the overall usefulness of jobs which are served, servers make decisions about which task to allocate based on the status inside as well as the features of requests that are incoming. An problem to enhance an online workload distribution system which authorizes for pre-emption while taking elastic resource requirements and deadlines into consideration was officially specified by the author. To increase the system's scalability, they also provided details on a clustering heuristic. A workload distribution mechanism for Enhanced Efficiency Across Multiple Devices and Base Stations for Minimizing Delay was suggested by the author in [16]. In intersecting domains of MEC, the goal is to propose an effective multidevice and multi-BS task offloading strategy to reduce time in job completion for Internet of Things devices. The article introduces the "DOLA" distributed work offloading method, which is based on the noncooperative game theory. In order to achieve a Nash equilibrium, DOLA seeks to decentralized task offloading decision-making optimization, making sure that no participant has an incentive to unilaterally modify its approach. The primary performance metric considered in this study is the reduction of work completion delay. The abstract emphasizes that these trials show the effectiveness of DOLA, with a focus on its greater performance as compared to alternative offloading techniques.

An Edge Computing method for distributed workload distribution in Smart Cities depending on the Internet of Things was also proposed by Omar Abdulkareem Mahmood et al. [19] to look at a multi-criteria optimization problem of a connected city which is based upon IOT, paying close attention to reducing the energy used and latency. In this study, a multi-layer consisting of 3 layers network topology was employed for connected cities. The first layer was the Internet of things and the second one was edge devices. Clouds made up the third stratum. With regard to range of virtual machines versus the

latency, the suggested model on average outperformed existing approaches by 3.1% with 90 VMs and 9.2% with 30 VMs. Additionally, with 200 jobs, advancement with regard to the quantity of jobs compared to the computational delay was 7%. Additionally, the average improvement in energy usage compared to the jobs was 157% assigned to 200 tasks, while the gains in energy usage compared to count of virtual machines were 188% with 30 VMs and 565% with 90 VMs. Kaige Tan et al. [7] suggested an optimized technique for jointly offloading the tasks and allocating the resources which is decentralized for Vehicular Edge Computing Systems. It was for systems that used mobile edge computing. Here, the issue is broken down into two smaller divisions, such as offloading of jobs and workload distribution at the RSU and vehicle levels. Roadside units, or RSU, are used here. Dual decomposition makes the resource allocation problem simpler and more amenable to decentralized solutions. A probability-based method converts the discrete task offloading problem into a continuous convex problem. On a 1-km roadway, they stimulated that. Due to its benefit, decentralized offloading performs load balancing at the highest level.

2.3 QoS Based Task Allocation

In [14], the author develops an QoS based DEP technique. With regards of EC for IoV, DEP seeks to strike a compromise between the quantity and QoS, as well as reconstruction cost of existing ES deployments. Comparative studies utilizing actual traffic data are used to demonstrate the efficacy of DEP. According to comparative studies using actual traffic data, DEP delivers a lower latency and a smaller workload standard deviation than the clustering technique by a margin of 17.64% and 25.82%, respectively. Using the current placement as a benchmark, the performance of the ES placement is evaluated in terms of average latency and load balancing. The outcomes show how DEP may effectively boost edge computing's functionality in the IoV. For CEC-IoV, here the author of the research [17] proposes a levelled design to ensure QoS and low power consumption workload distribution, and latency and how efficiently the energy is used at Mobile edge computing systems are optimized, correspondingly. The assignment problem is resolved using a combinatorial optimization strategy in a polynomial amount of time. A suitable reaction time threshold is used to detect underloaded and overloaded MECSs, and load-balancing is then carried out among them. Additionally, VMs at a MECS have workload redistribution and computing resource reconfiguration integrated.

Additionally, the paper [3] intended to raise EC's level of service quality by using a game theoretic technique to handle load imbalance problems in linked IEC. A learning algorithm which was decentralized was suggested. Each IES initially chooses scheduling actions based on their strategies, which they then review and update to attain Nash equilibrium. This iterative process continues over many phases, dynamically adjusting task allocation. The suggested state-based gaming strategy outperforms the FMBRID technique and provides load balancing that is comparable to the optimization approach, especially in settings with bursts of task arrivals. The authors' goal was to maximize user Quality of experience by concurrently enhancing the selection of service, distributing the workload and offloading the tasks decisions by proposing a method which is distributed namely Lagrangian-dual based decomposition theory [8]. It demonstrated NP-hardness for the assignment by formulating it as a combined-integer nonlinear programming

issue. It provided resource-efficiency based heuristic after reformulating the issue as a NUM problem in order to solve it effectively. Finally, it tested our methods using several simulations, and the outcomes showed how effective they were. It has been found that under real-world task demands, the cost-aware online algorithm outperforms the Highest Allocation of Rate method by more than 50% while requiring fewer service changes.

2.4 Energy Efficient Based Task Allocation

Tomaso Erseghe's [2] aims to make edge computing more energy-efficient and sustainable in the era of connected devices. It emphasizes workload distribution, reduced non-renewable energy use, load balancing, and server consolidation. The aim is to enhance sustainability, cut costs, and reduce grid dependency. Renewable energy for edge servers is encouraged while ensuring timely task completion. To predict arrivals, correlated Markov Chains with ON-OFF behavior are used. Three prediction methods, including the Genie Predictor (ideal), support Model Predictive Control (MPC). MPC leverages System State Equations for offloading and the Energy Consumption Model to optimize workload scheduling, energy management, and server decisions. It focuses on energy buffers. MPC uses two cost functions, quadratic and logarithmic, to minimize resource allocation costs, promoting server consolidation and load balancing. This leads to two optimization problems: one convex and one non-convex. In this paper, a decentralized MPC-based job allocation strategy for MEC networks is presented. It outperforms heuristics and myopic approaches, reduces grid dependency, and paves the way for GPU energy models and user mobility-aware workload distribution.

The objective of the research article [4] is to optimize mobile-edge computing for enhanced user experience and cost-efficiency by reducing energy consumption, task response delay, and cloudlet deployment. It introduces a powerful optimization algorithm. MGW, a multi-objective optimization technique, is used in Mobile Edge Computing (MEC) systems for job offloading and cloudlet distribution. In MGW, a collection of non-dominated solutions are discovered by combining a whale optimization algorithm with guided population archiving methods. The problem is NP-complete and has multiple objectives, and work on heuristic algorithm to get better solution. For enhancing discovery, MGW also uses opposition-based learning. According to simulation data, MGW acheives better solution when compared to other algorithms in contexts of solution quality and diversity as assessed by inverted generational distance (IGD) and hypervolume (HV), making it a useful tool for MECS service providers. MINP (Mixed-Integer Nonlinear Programming) is the method used to formulate the objectives. This NP-complete problem is addressed using a modified version of the GPAWOA (Gravitational Search Algorithm with Opposition-Based Learning), which improves position repair, initialization, and encoding techniques. The algorithm's performance is enhanced by incorporating a variety of optimization approaches, including GOBL, QOBL, and mutation/crossover from Differential Evolution. The suggested algorithm is more effective than existing metaheuristic algorithms at producing high-quality nondominated solutions quickly. Future work will entail applying multi-objective optimization to optimize channel bandwidth allocation.

The paper's [11] objective is to create a load-balancing group for container distribution synthesis, and migration in a SDE environment that uses little energy. Through simulation, this method is confirmed by looking at performance indicators such as energy usage, network, CPU deliver times, and the final retard time. Goal is to offer a SDE solution for delicate, CaaS for IoT applications that are latency-sensitive. In order to maximize performance and energy consumption, the suggested algorithm uses multi-layered system modeling, a multiple leader and follower game namely, Stackelberg and container-derived from virtualization. The results show decreased energy consumption in comparison to existing variations, regardless of changes in workload size.

2.5 Resource Aware

Resource-aware task allocation refers to the distribution of workloads according on the processor, memory, and battery life currently available on edge devices. The authors in [6] presented a messaging system for Artificial Internet of Things in edge computing (a distributed method) which concentrates on message ordering issues. DMSCO technique, which is dependent over DDPG, is also recommended in this study in order to enhance the efficiency of messaging system. The outcomes displays profound influence of the suggested DMSCO approach, delivering an exceptionally efficient output of 88.79 MB/s, which is an outstanding enhancement of 46.61% exceeding the messaging system (distributed) without any optimized configuration. Random searching, in contrast, shows a 22.17% improvement.

Table 1. An overview of edge computing task distribution techniques.

Ref.	Motto	Proposed Algorithm	Workload Distribution technique	Networks Utilized	Advantages	Disadvantages
[1]	Service pricing and workload allocation in cvec	Optimal contract design under prospect theory	AI	Vehicular	User-Centric Utility Maximization, Economic Benefits	Scalability Challenges, Resource Variability
[2]	Enhance energy-efficiency, reducing grid dependence, and optimizing edge task management	MPC Based Allocation of Processing Tasks	Energy-efficient	EC	Reduces the Transmission costs and renewable energy sold. Adapting to changing conditions	User Mobility is a challenge for handling of dynamic user movement. Can't address the complexity tasks
[3]	Improve QoS in EC by using game theory to balance IEC load	State-Based Decentralized Learning Algorithm	QoS	IEC	Enhance the Service Performance. Getting into optimal state for each reachable state	Couldn't address the load balancing issues with multiple types of servers under wireless connections

(*continued*)

Table 1. (*continued*)

Ref.	Motto	Proposed Algorithm	Workload Distribution technique	Networks Utilized	Advantages	Disadvantages
[4]	To reduce the amount of cloudlets deployed, task response time, and energy usage	Whale Optimization Algorithms 1. RFSCA 2. Sr Algorithm MGW algorithm	Energy-efficient	IOT	Energy and Delay Reduction, Multi-objective Optimization, Algorithm efficiency and quality for effective offloading	Fails to maximize channel bandwidth distribution, testing multi objective optimization algorithm
[5]	A method to boost serverless computing effectiveness in edge contexts	Genetic Algorithm (GA) Integer Linear Programming (ILP) Hidden Markov Model (HMM)	Ai	EC	Optimization Effective use has been made of factors like task costs, computational capacity, availability, and time restrictions	Lacks in improving of serverless function placement methodologies and improving in dynamic workload distribution
[6]	To address the challenges associated with message ordering	PSA, DMSCO	Resource Aware	IOT	Carries out distributed messaging system auto-configuration in edge contexts for the Artificial Internet of Things	Manual selection is done for picking between numerous versatile factors and important parameters might be missed out
[7]	Task offloading and resource allocation joint solution	Decentralized convex optimization	Distributed	Vehicular	Replaces the initial binary decision problem by a constantly optimizing	System's reliability is not investigated
[8]	An optimized solution for workload distribution and task offloading to maximize user quality of experience	Lagrangian-dual based decomposition theory-based distributed algorithm	QoS	MEC	Good solution for MEC systems with scarce resources	Online algorithms may not perform as well as offline algorithms in terms of optimizing certain criteria because they are not aware of the entire input

(*continued*)

Table 1. (*continued*)

Ref.	Motto	Proposed Algorithm	Workload Distribution technique	Networks Utilized	Advantages	Disadvantages
[9]	Balancing workload in edge computing	Deep-Q-Network (DQN)	AI/ML	EC	Improves performance by balancing workloads and reduce failed tasks rate	The approach may not adapt to dynamic situations and the resource efficiency is not good
[10]	Mobile user reallocation using the MEC service in busy environments	Coalitional Game-Based Service Migration	ML	MEC	High efficiency in matter of resource use, load balancing, and the number of necessary service migrations	The proposed method's performance may limit its usability in less congested environments
[11]	To provision the workloads produced by the IoT apps that require low latency	Container-based virtualization	Energy-efficient	Other	The results obtained align superior compared to standard approach	This involves multiple components resulting in increased complexity in terms of management and resource consumption
[12]	The deployment of MEC and resource management are made easier	Spatio-temporal Bayesian hierarchical learning approach	AI	MEC	The outcomes are more effective than distributing resources equally among all of the servers in unseen areas	High complexity
[13]	Reducing its net cost while offering edge computing services	Collaborative queueing game strategy with novelity	AI	MEC	Suggested method gives long time optimal and equilibrium solution	Practical ability of this model may be limited due to the assumptions made
[14]	Proposing dynamic edge server (ES) placement approach for IoV in the intelligent transportation system (ITS)	DEP (Dynamic ES Placement)	QoS	Vehicular	Perform better and require less reconstruction of current placement of servers	Less latency and less variation in workload

(*continued*)

Table 1. (*continued*)

Ref.	Motto	Proposed Algorithm	Workload Distribution technique	Networks Utilized	Advantages	Disadvantages
[15]	Optimizing function placement in serverless edge computing systems	NSGA-II	AI	IOT	Flexibility, extensibility, operational goal optimisation, and precise resource modeling	Very low Resource utilization
[16]	An efficient task offloading scheme to minimize delay and improve performance	"DOLA" based on the theory of noncooperative game	Distributed	IOT	MEC delivers benefits like less communication overhead and delay, greater scalability and work-balance	Restrictions on the material's availability, depth of detail, reliance on other sources
[17]	Hierarchical Resource Management Model for Cooperative Edge Computing in IoV	Hungarian Algorithm	QoS	MEC	Lowering power consumption, raising service quality	The optimization is not carried out cooperatively
[18]	Utilizing the PSO method for organizing application modules created by the edge network's workload effectively	BMOPSO	AI	IOT	Improve the reliability and reduction of errors	High latency, bandwidth utilisation, power consumption
[19]	To investigate a multicriteria optimization issue in an IoT-based smart city, specifically focusing on minimizing energy consumption and delay	MRFO, SSA, HHO	Distributed	5G	Decreased network delays, increased QoS, and stability	Scalability, performance, security, costs
[20]	Distributing and assigning edge services through a distributed method	Two-round bidding approach and clustering	Distributed	MEC	Accelerates and makes the auctions more scalable by using clustering heuristic	The Servers need to have knowledge of job utilities

3 Quantitative Analysis

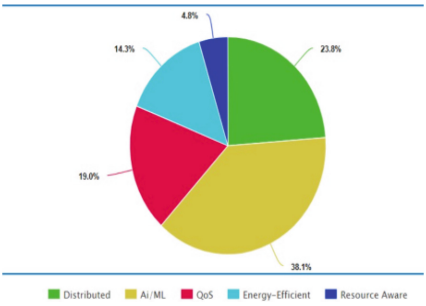

Fig. 2. Amounts of the various task distribution techniques utilized.

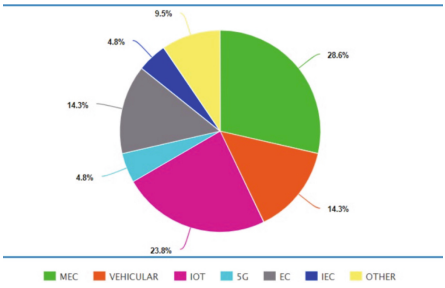

Fig. 3. Proportions of communication networks utilized in task allocation.

In the analysis of the literature, the utilization of various technologies for the purpose of task allocation and communication networks was inquired. The graph Fig. 2 depicts the task allocation approaches covered in the publications we read. Especially, artificial intelligence (AI) and machine learning (ML) task allocation methods were exclusively used representing 38.1% of surveyed papers. This dominance underscores the importance of AI/ML in optimizing task distribution. Also, the distributed task allocation an emerging well-known methodology representing 23.8%. This shows the continuous relevance in decentralized systems, where nodes collaborate freely to allocate tasks efficiently. The task allocation based on the Quality of service (19.0%) which shows the importance of prioritizing tasks to meet up with predefined performance, which ensures the user experience. Energy efficiency (14.3%) is another considerable method, with strategies aiming to minimize energy consumption while ensuring the maintenance of computational tasks, Further, the Resource awareness (4.8%) which highlights the importance of prioritizing available resources while making decisions on allocation which facilitates the optimized resource utilization and also system performance. All these findings signifies the diversity of the various task allocation strategies employed in research.

Figure 3 depicts the many types of communications networks utilized in the literature. MEC was the most extensively suitable network type, cited in 28.6% of the analyzed papers. This shows the increasing in adoption of MEC to facilitate low-latency and high-bandwidth application. IOT networks followed up by 23.8%, which highlights the integration of IOT devices and interconnected smart devices. Moreover, the existence of vehicular and electrical networks, representing 14.3% of each which specifically signified the importance of transportation and power distributed systems. While 5G (4.8%) as it is a new emerging technology and IEC (4.8%) were less frequently referenced. The remaining percentages are represented by the category various types of networks (9.5%).

4 Conclusion

The task of distribution of work in edge computing is complex and time consuming. It needs a careful examination of each application prerequisites. In this comprehensive analysis of task allocation techniques in edge computing, a number of methodologies for assigning work on edge intelligence devices, are studied and evaluated including decentralized, centralized, and heuristic machine learning algorithms. The main important issue in task allocation is to determining the best placement for each activity based on power, privacy, requirements of the bandwidth, as well as it should be able to adapt and adjust to the changing nature of the network. This review is based on the investigations and the study on task analysis, Internet of Things, energy efficiency, privacy and intensive communication workloads are some of them for understanding a range of objectives. In more detail, we categorized the various workload distribution strategies used, as well as the network types for the best task offloading. The significant importance of distribution of tasks in edge computing for IoT devices, AIoT environments, industrial applications, connected vehicles, smart grids, UAVs, and IoV, among other applications, is strongly supported by our review.

References

1. Huang, X., Yu, R., Ye, D., Shu, L., Xie, S.: Efficient workload allocation and user-centric utility maximization for task scheduling in collaborative vehicular edge computing. IEEE Trans. Veh. Technol. **70**(4), 3773–3787 (2021)
2. Perin, G., Berno, M., Erseghe, T., Rossi, M.: Towards sustainable edge computing through renewable energy resources and online, distributed and predictive scheduling. IEEE Trans. Netw. Serv. Manage. **19**(1), 306–321 (2022)
3. Zhang, F., Deng, R., Zhao, X., Wang, M.M.: Load balancing for distributed intelligent edge computing: a state-based game approach. IEEE Trans. Cogn. Commun. Netw. **7**(4), 1066–1077 (2021)
4. Zhu, X., Zhou, M.: Multiobjective optimized cloudlet deployment and task offloading for mobile-edge computing. IEEE Internet Things J. **8**(20), 15582–15595 (2021)
5. Zhang, R., Shu, H., Navaei, Y.D.: Load balancing in edge computing using integer linear programming based genetic algorithm and multilevel control approach. Wirel. Commun. Mob. Comput. **2022**, 1–22 (2022). Article ID 6125246
6. Xie, Z., Ji, C., Xu, L., Xia, M., Cao, H.: Towards an optimized distributed message queue system for AIoT edge computing: a reinforcement learning approach. Sensors **23**, 5447 (2023)
7. Tan, K., Feng, L., Dán, G., Törngren, M.: Decentralized convex optimization for joint task offloading and resource allocation of vehicular edge computing systems. IEEE Trans. Veh. Technol. **71**(12), 13226–13241 (2022)
8. Chu, W., Yu, P., Yu, Z., Lui, J.C.S., Lin, Y.: Online optimal service selection, resource allocation and task offloading for multi-access edge computing: a utility-based approach. IEEE Trans. Mob. Comput. **22**(7), 4150–4167 (2023)
9. Zheng, T., Wan, J., Zhang, J., et al.: Deep reinforcement learning-based workload scheduling for edge computing. J Cloud Comput. **11**, 3 (2022)
10. Xiao, X., et al.: Novel workload-aware approach to mobile user reallocation in crowded mobile edge computing environment. IEEE Trans. Intell. Transp. Syst. **23**(7), 8846–8856 (2022)
11. Singh, A., Aujla, G.S., Bali, R.S.: Container-based load balancing for energy efficiency in software-defined edge computing environment. Sustain. Comput.: Inform. Syst. **30** (2021)

12. Ale, L., Zhang, N., King, S.A., Guardiola, J.: Spatio-temporal Bayesian Learning for Mobile Edge Computing Resource Planning in Smart Cities. ACM Trans. Internet Technol. **21**(3), 1–21 (2021). Article 72
13. George, C.M., Sharma, D., Reeja, S.R.: Mobility prediction-based source anonymity routing protocol (MPSARP) for source location privacy using NS2 techniques. J. Theor. Appl. Inf. Technol. **101**(9) (2023)
14. Shen, B., Xu, X., Qi, L., Zhang, X., Srivastava, G.: Dynamic server placement in edge computing toward internet of vehicles. Comput. Commun. **178** (2021)
15. Rausch, T., Rashed, A., Dustdar, S.: Optimized container scheduling for data-intensive serverless edge computing. Future Gener. Comput. Syst. **114** (2021)
16. Huang, J., Wang, M., Wu, Y., Chen, Y., Shen, X.: Distributed offloading in overlapping areas of mobile-edge computing for internet of things. IEEE Internet Things J. **9**(15), 13837–13847 (2022)
17. Duan, W., Gu, X., Wen, M., Ji, Y., Ge, J., Zhang, G.: Resource management for intelligent vehicular edge computing networks. IEEE Trans. Intell. Transp. Syst. **23**(7), 9797–9808 (2022)
18. Rodríguez, O.R.C., Le, V.T., Pahl, C., Ioini, N.E., Barzegar, H.R.: Improvement of Edge Computing Workload Placement using Multi Objective Particle Swarm Optimization. In: 2021 8th International Conference on Internet of Things: Systems, Management and Security (IOTSMS), Gandia, Spain (2021)
19. Muneeswari, G., Reeja, S.R.: Agent based queue aware scheduling for distributed multicore system. In: 2022 IEEE 7th International Conference on Recent Advances and Innovations in Engineering (ICRAIE), Mangalore, India, pp. 1–6 (2022). https://doi.org/10.1109/ICRAIE56454.2022.10054343
20. Rublein, C., Mehmeti, F., Gunes, T.D., Stein, S., La Porta, T.F.: Scalable resource allocation techniques for edge computing systems. In: 2022 International Conference on Computer Communications and Networks (ICCCN), Honolulu, HI, USA (2022)