A Survey: Resource Allocation Technologies based on Edge Computing in IoT

LAKSHMANAN S[1]
Research Scholar
Department of Computer Science
St. Joseph's College (Autonomous),
Affiliated to Bharathidasan University,
Tiruchirappalli, Tamil Nadu, India.
laxmphil@gmail.com

Dr.T.KOKILAVANI[2]
Assistant Professor
Department of Computer Science
St. Joseph's College (Autonomous),
Affiliated to Bharathidasan University,
Tiruchirappalli, Tamil Nadu, India.
. kokilavani77@gmail.com

***Abstract*— The Internet of Things (IoT) is a new prototype that connects a range of devices to the Internet through wireless and wired technologies. The processing of this data necessitates early processing, resource allocation, response, an increase in transmission latency, and performance concerns for IoT devices. Edge computing is a novel technology that has been developed to address these issues. Edge computing refers to information processing on a device, a local computer, or a server. The edge servers are located close to the user of IoT devices and serve as a link between the cloud and the user. This technology reduces data transfer delay and improves the device's speed performance. System execution, device utilization, memory management, and network bandwidth are all improved. Security, privacy, Internet walls, and cloud attacks are all-new concerns for IoT resource allocation. As a result, an IoT-specific resource allocation technique should be developed. Various technologies and techniques used in edge computing for resource allocation are explored in this overview. It primarily focuses on the dynamic allocation of various edge servers or devices, as well as the benefits and important difficulties encountered in this environment, simulated procedures, parameters, and future development.**

***Keywords*— Internet of Things (IoT), Edge Computing, Mobile Edge Computing, Cloud Computing, Resource Allocation**

I. Introduction

1.1 An Overview of Edge Computing

Cloud computing is a distributed computing system that stores and processes data. Latency, bandwidth, and security access are among the characteristics that are restricted. To address this issue, new edge computing technologies have been launched. The need for the Internet of Things and IoT devices is booming these days. Cloud computing collects a significant amount of data and stores it on centralized computers in data centres. Data is created on the end device and then sent to a cloud server to be processed. Because the fundamental issue here is delay [2] this becomes costly for procedures that require demand.

When the data is closer to a user, it will be shared with them quickly, securely and also without latency. The combination of cloud and edge will improve consistency and response time. Edge computing is a technology in which hardware and software are physically located at the network's edge. Cloud computing performs workloads within clouds, whereas edge computing performs workloads on edge devices [3].
Edge computing processes include data evaluation, management, and movement at the network's edge. In other words, the data is evaluated locally, where it is connected rapidly to the edge device or server. The evaluated data is saved in the cloud indefinitely.

1.2 Benefits and Drawbacks of Edge Computing

Benefits:
- It provides high speed, reduced latency better reliability which allows for faster data management and content distribution.
- It provides security by processing, storage, and requests across a broad range of devices and data centers.
- It also provides a cheaper route to scalability and flexibility, allowing firms to enlarge their calculating capacity through various IoT devices and edge data centers [5] [6].

Drawbacks:
- Edge computing has significant security challenges due to the increased volume of data.
- It only works with data.
- The cost of edge computing is really high.
- It necessitates sophisticated infrastructure [5] [6].

1.3 Edge Computing Architecture:

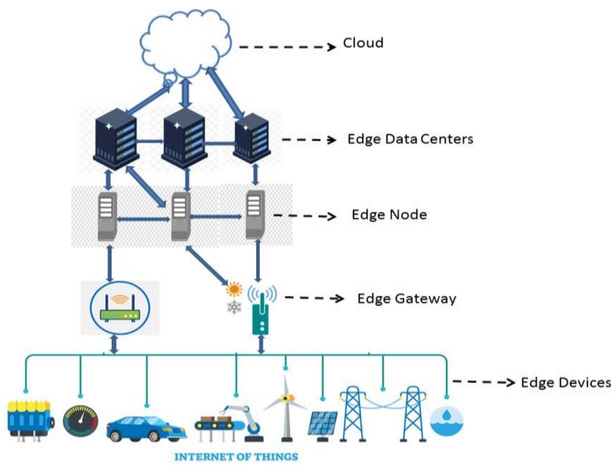

Figure 1. **Edge Computing Architecture**

Figure 1 shows the architecture of edge computing.
The various processing element of edge computing are:
Cloud: The computational power and loading capacity of the cloud layer is almost limitless, but data transfer costs and latencies are quite high. Edge computing applications can use the cloud as persistent storage. It also handles powerful resources for tasks that aren't balanced.

Edge Node: Edge nodes or "downstream" nodes are positioned above the last mile of the network and are also called as gateway nodes or edge communication nodes. It's a machine that acts as a communication portal for users of various cluster computing systems. This device has enough processing power to route network traffic. They can range from small data centres to base stations, routers, and switches.

Edge Gateway: Edge Gateways are similar to edge nodes but are less powerful. It's a standard protocol that can handle computations without specific hardware. Edge Gateway has devices; it enables network translation for lower-level devices like mobile phones and automobile sensors to disseminate (cameras and motion detectors).

Edge Devices: Edge Devices are small devices with few resources, such as a single sensor. These devices have limited communication capabilities and are employed for a specific sort of computation. It is a device that provides a point of entry into the service provider's basic networks and operates on very simple principles. It also serves as the network's entry and exit points. Smart watches, traffic lights, and environmental sensors are all examples. Routers, routing switches, integrated access devices (IADs), multiplexers, and a variety of metropolitan area network (MAN) and wide area network (WAN) access devices are also included in the edge deriver. Edge devices connected to an edge node have no limitations, and numerous edge nodes in different places can be connected to a single edge node [2] [3] [4] [7].

1.4 Working of Edge Computing

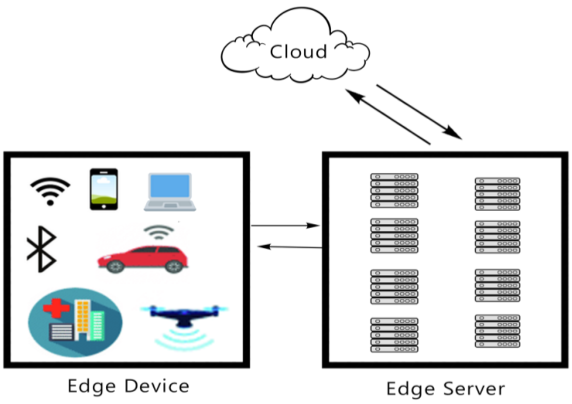

Figure 2. **Working on Edge Computing**

Fewer processes are moved from the cloud to local locations, such as a user's PC, an IoT device, or an edge server, with edge computing. Edge computing, in other terms, is a distributed IT architecture that delivers computing resources from clouds and data centres as close as feasible to the source. Its main goal is to reduce latency when processing data and cut network costs. The most crucial aspect of this network edge is that it should be close to the device geographically [2] [3] [4] [7].

In edge computing techniques, data is first displayed on a user's screen before being communicated to a server via the internet, intranet, LAN, and other means. The data is kept and processed on the server. However, as the amount of data and the number of devices linked to the internet grew, these data storage infrastructures began to appear incapable of keeping so much data. According to a Gartner study, 75% of enterprise data would be created outside of the centralized data centres by 2025. This massive amount of data puts a huge strain on the internet, producing congestion and outages. The concept of "Edge Computing" was proposed to address this. The concept behind edge computing is simple: instead of moving data closer to the data centre, the data centre is brought closer to the data. The storage and processing resources in the data centre are placed as close as feasible to where the data is created (preferably in the same place). An edge gateway, for example, can process data from an edge device, which is a device that controls the flow of data between two networks, and then send just the data that is required back to the cloud, reducing bandwidth requirements. It can also send data back to the edge device in the case of real-time application requirements. Edge devices include IoT sensors, employee notebook computers, smartphones, security cameras, and even the internet-connected microwave oven in the office break room. Edge gateways are considered edge devices in an edge computing environment.

1.5 Resource Allocation for Edge Computing

Artificial intelligence, cloud computing, Edge computing, Fog computing, and Machine learning are all examples of distributed computing technologies. The primary purpose of IoT devices is to generate data, turn usable information, and provide resources to users. In this scenario, IoT resource allocation and scheduling is a critical study field. It must deal with challenges such as dynamic resource allocation, improved system performance, service quality, service level agreements, and data cleanup. The Internet of Things (IoT) and edge computing are significant resource allocation research problems. As a result, the algorithm's resource allocation dynamic model and the foundation of network/server-status are built. The fundamental goal of the algorithm is to alter and improve dynamically to produce integrated communication, processing, storage, and virtualization. During resource allocation, the following issues arise: (i) Allocate storage, compute, and communicate resources. (ii) Changing network topology/protocols; and (iii) arranging edge computing data centres (VM distribution). Static and dynamic resource allocations are the two types of resource allocation [21].

1.5.1 Allocation of Static Resources

Static resource allocation in edge computing refers to allocating existing resources to fixed-edge apps through the internet/intranet. If resource allocation is not managed correctly, services will suffer. Once specified resources have been allotted to a job, it cannot be altered while it is being executed. The element that references static resources only evaluates them once during task loading. It is quick and saves time, but it is inefficient compared to the dynamic allocation technique [21].

1.5.2 Allocation of Dynamic Resources

In this allocation, resources are allocated without human intervention in varied situations based on task needs. Dynamic resource allocation in edge computing is the process of allocating existing/new resources to edge applications over the internet/intranet without requiring human input. It can be adjusted when work is being executed, with the assistance of the administrator, who will assign all resources to the various jobs. The resources that users can change and analyze at runtime are known as dynamic resources. It is a little slower than the static allocation policy, but it is more efficient. This form of allocation is difficult to implement [21].

II. REVIEW OF LITERATURE

Xiancui Xiao et al., [1] described that, Edge computing is a critical component of network resource deployment and allocation. In edge computing contexts, resource allocation technology faces the difficulty of virtual network mapping. Existing solutions are limited to static resource allocation, avoiding time-varying jobs; however, user resource demands vary over time, and resource allocation using analytical methods is a promising solution. Now, based on the group search optimizer (GSO) and incremental RBF design, a dynamic network resource demand prediction algorithm is introduced (GSO-INC-RBFDM). The GSO boosts the server solution and validates the user's dynamic resource requirements. The incremental design is also used to eliminate the maximum error value. When compared to existing Greedy, MCF algorithms, the proposed mechanism enhances acceptance rate, network cost, link weight, and average income.

Ying He et al. [7] proposed a centralized Blockchain-based edge computing resource allocation technology that can be done through locally and remotely. Between IoT devices and edge computing nodes, there are security and privacy concerns. To address the above-mentioned challenges, the proposed mechanism used Blockchain technology and a deep learning technique, as well as designing the state-of-the-art machine learning algorithm, A3C (Advantage actor-critic), and combining AI and Blockchains. Future work for collaborative optimization of Blockchain parameters and edge computing resource allocation was proposed by the author.

Youngjin Kim et al. [8] stated that, an “IoT assisted edge computing". It's implemented on an edge server with a task-supported completion time-based task and a hierarchical weight allocation system that uses IoT devices as task schedulers. It can increase the throughput of edge services and ensure that local IoT task deadlines are met. The above method to handle the rich computational resources is becoming increasingly relevant, since some devices were not functioning properly and an underutilized resource in IoT devices for other expressive work on the local IoT task was becoming increasingly vital.

Abdullah Khanfor et al., [9] stated that, the Internet of Things (IoT) network's diversity was broken as a dynamic computational resource environment for numerous devices that lack computing capability. The new framework was proposed using machine learning techniques and it was divided into two phases. The two phases are as follows:

1. Trustworthy Community Detection Phase (TCDP) – the ultimate goal is to find communities of IoT devices that have strong social relationships. The goal of this step is to locate appropriate device spaces and minimize the computation task to reliable devices.

2. Matching Device Phase (MDP) - The goal of this phase is to predict the time necessary for potential edge computer users for the same client community to complete and regulate the activity.

The proposed technique simplifies the service discovery effort, potentially and to the search minimizing the search space, which is beneficial for large-scale IoT systems.

Jiechen Chen et al., [10] proposed a Multi-user caching-enabled (MEC) system, in which users submit task requests repeatedly and the edge server executes them. It is possible to minimize the weighted-sum energy of the edge server by applying the suggested methodology for Joint Cache Placement and Bandwidth Allocation for FDMA-based Mobile Edge Computing Systems to handle this mixed-integer non-convex problem and BW allocation problem using the ellipsoid technique. The heuristic algorithm changes the cache placement regularly, reducing complexity and improving task performance.

Xiaolan Liu et al., [11] proposed an IoT application which includes edge computing, centralized cloud computing with distributed data processing, and low latency. It solves the problem of computing offloading in IoT edge computing networks with conflicting resources such as transmit power, radio access technology, sub-channel, and multi-user IoT edge computing networks. The proposed framework for Multi-Agent Reinforcement Learning for Resource Allocation in IoT Networks with Edge Computing, which uses the proposed Independent Learners-based Multi-Agent Q-learning (IL-based MA-Q) algorithm was used to solve the mobile game. The IL-based MA-Q algorithm was used for various sizes of IoT networks and the simulation results were confirmed. It also solve radio access collisions, minimized system costs, and increase multi-user IoT networks and computing performance by running in parallel with the three benchmark methods.

Sasa Pesic et al., [12] stated that the edge computing was used to improve the scalability, reactivity, efficiency, and privacy of IoT systems. IoT systems were complicated, and the task of moving in the direction of managing resources and service supply was difficult. Resource management of context-aware decision-making processes distributed between IoT gateways was a major task for edge computing. Edge computing is a system for managing decision-making processes, monitoring sensory data streams, delivering context-aware edge computing resources, and managing operational changes through service provisioning management. Now that the RMF (Resource Management Framework) framework has been deployed on edge computing, the responsible operation has been done and validated in response to topology changes. The new framework addresses the recovery of failed decision processes, which had an impact on the IoT system's health.

Qianjun Wang et al., [13] explained the unmanned aerial vehicle (UAV) inspection edge computing was used to collect data on field operation control, power consumption, and the power Internet of Things scene. The processing, low latency services ensure that service qualities are all edge computing offload functions. Because the edge node issues were limited and making it difficult to meet the criteria for power IoT and job allocation at the same time. It was stated a new work assignment mechanism based on cooperative edge computing. Two edge nodes working together have been shown to reduce average job completion time while meeting business requirements. The Two-edge-node Cooperative-task Allocation technique based on Improved Particle Swarm Optimization (TCA-IPSO) was then proposed. The TCA-IPSO algorithm resolves the job allocation scheme. The author introduced a new hybrid algorithm that occurs simultaneously with an optimization algorithm to improve the efficiency of algorithm search.

Min Chen et al., [14] described that, the 5G mobile device communication networks. The network component boosts mobile traffic, which is crucial for 5G mobile device resolution. 5G mobile devices are covered by communication applications such as enhanced mobile broadband (eMBB), ultra-high reliability, and ultra-low latency (uRLLC). The foregoing factors, when combined, alleviate several concerns such as uRLLC latency and reliability requirements. Artificial intelligence has been used to govern the flow of mobile traffic in the proposed mechanism. It distinguishes between two sorts of site modes: single-site mode and multi-site mode. This mechanism sequences the mobile traffic data and uses a Long Short Term Memory-based user traffic flow prediction algorithm (LSTM). By forecasting the peak value, the aforesaid mechanism and algorithm successfully solve the traffic flow. Multiple-site scenario: It is built on an IoT-based mobile traffic prediction-and-control architecture that can be dynamically allocate communication and compute resources. It can be accomplished by lowering communication latency and the packet-loss ratio. Our intelligent mobile traffic-flow control system, which is based on an IoT Cloud, will be tested in the future.

Jianhui Liu et al., [15] elicited an, important mobile edge computing (MEC) technology which included various types of heterogeneous services within the MEC system. Depending on the service, the task arrival interval and execution time change. Virtual reality (VR), augmented reality (AR), and autonomous vehicle applications are among the computationally expensive jobs emerging for the Internet of Things (IoT) with minimal latency. It deals with scheduling computing resources at a location edge server depending on service arrivals and runtime (ES). Based on a framework integrated with the Lyapunov-based algorithm, the author proposed a framework for computing offloading between users and ESs. On ES, it can be dynamically assigned for a variety of time-critical services. The new techniques, which effectively perform the computation resource, reduce the average timeout for probability without prior knowledge of the task arrival process and needed runtime.

Zhang Bo-wei et al., [16] proposed the combination of edge computing and wireless power transmission. It is a piece of low-suspension information and sustainable green energy IoT process. This application addresses concerns with IoT system performance that are caused by main energy consumption considerations. A time-division, duplex-orthogonal frequency division multiple access modes proposed an energy-efficiency-based resource allocation and power regulation algorithm (EE-RAPC) to Wireless Power Transfer and Edge Computing system (WPT-EC-IoT) (TDD-OFDMA). An algorithm optimizes channel allocation based on subchannel priority at first. The terminals consider the channel with the highest priority. The power control system also assigns varying levels of power to subchannels. The channel Allocation Algorithm and Power Control Algorithm were used to achieve increased IoT terminals and average task offloading, as well as optimal performance.

Juan Fang et al., [17] described that, the data is processed in real-time by IoT devices and saved in the IoT Cloud. It solves problems with typical cloud computing network topologies. Edge computing offloads services to the edge server and performs the close terminal device using cloud computing. It addresses the major issue of lowering cloud power usage and processing load. In heterogeneous network environments, the proposed mechanism is to resolve multiple task demands of IoT devices. To provide a mapping mechanism between application modules and basic resource equipment that takes into account task delay and power consumption tolerance. The heuristic method is implemented step by step in a task processing application that is extremely dynamic. It utilized to cut down on task delay. When the "iFogSim" simulator is used in an application, it is possible to reduce system power consumption and improve the quality of the applied service by comparing static application and static task scheduling strategies. The difficulty of dynamically changing the mapping of application resource nodes and task execution during task execution to satisfy the needs of diverse IoT application services.

Amit Samanta et al., [18] proposed a dynamic microservice scheduling solution for MEC to deal with the problem of overall network delay and network pricing using microservice schedules. To introduce the microservice scheduling framework, this is conceptually constructed and make use of the computational complexity of the scheduling algorithm. The novel microservice scheduling design improves total network latency, average price, satisfaction level, energy consumption rate (ECR), failure rate, and network throughput. Internet-of-Things (IoT) applications are efficiently executed at the network edge in the Mobile Edge Computing (MEC) environment. The environment is the best resource for various microservices; it was influenced by network constraints and infrastructures during execution, requiring acceptable Quality-of-Service, and enhancing energy efficiency in edge servers (QoS).
The author suggests that future work should focus on developing an actual system or prototype for dynamic microservice scheduling with various fine-grained optimizations, as well as addressing the issue of adaptive microservice scheduling in edge-based healthcare systems.

Beomhan Baek et al., [19] provided a comparison of three dynamic pricing strategies for resource allocation in the edge computing and IoT environments, including the BID-PRoportional Allocation Mechanism (BID-PRAM), UNIform PRIcing Mechanism (UNI-PRIM), and FAIrness-seeking Differentiated PRIcing Mechanism (FAIDPRIM). Personal IoT data generated by the Internet of Things; its low-latency and security challenges, pricing systems, computing resource allocation of edge computing servers, and fairness difficulties. The BID-PRAM is a technique for avoiding auction-based pricing. A non-cooperative game is used to describe this technique. Both the UNI-PIM and FAID-PRIM mechanisms solve Nash equilibrium (NE) or Stackelberg equilibrium (SE) with proof of existence and uniqueness as a single-leader-multiple-followers Stackelberg game. The proposed comparative analysis of three ways to administrate service provider guidance on various types of pricing schemes in edge

eISSN 1303-5150

computing, including the merits and drawbacks of various models.

Xin Tang et al., [20] described that, Mobile Edge Computing (MEC); it provided a variety of resource architectures at the network edge. Edge services enabled enormous mobile and IoT devices to have AI capabilities. AI applications based on deep neural networks (DNNs) are run on an edge server with the user equipment. The UEs, which are resource-hungry and compute-intensive, are affected in real-time. Each edge server has a certain amount of resources, making dealing with difficulties in nature a resource-constrained optimization challenge. The above challenges to reduce using new framework for multi-user DNN partitioning and computational resource allocation. The techniques scale effectively by lowering the maximum delay among all UEs and attaining system optimum with polynomial time complexity. Consider the difficult scenarios of cooperative computation, communication, and energy resource allocation for multi-user edge intelligence applications in future research.

Jianxi Wang et al., [21] proposed that, the cloud edge computing environments, presented an optimization approach for computing resource allocation of large IoHT devices. The technology for 5G and Internet of Health Things devices is continually growing. The computer resource allocation solves the problem of existing technology's long delays and poor security performance. The 5G heterogeneous cloud edge computing network was built in this setting. The network status is assigned to local or edge computing depending on-device computing, while the server provides options for adjusting the computing delay, communication, and computing resource allocation. It addresses the threat of large-scale privacy data leakage in IoT by optimizing network computing delay and energy consumption with resource allocation as a goal, to sort the priority of subtasks and to understand the most effective allocation of computing resources and designing instant messaging privacy data. The local area network is used to connect the terminal devices. Without cloud servers, these local networks link to edge servers via sockets. The method described above enhancement and the security of personal information. The MATLAB simulation platform improved the performance of the allocation method by increasing the number of edge computing servers, users, device computing capability, and task arrival rate.

Xiong Xiong et al., [22] proposed the IoT edge computing system, a new resource allocation policy. This large volume of data can generate more IoT devices; the data is processed and analyzed at the network edge utilizing a mobile edge computing (MEC) environment. Virtual resources are limited in MEC systems; these resources are shared and competed in IoT edge applications. Using the Markov decision process (MDP) paradigm, this application expressed the resource allocation problem in an IoT edge computing system. The proposed policy improved resource utilization by lowering the long-term weighted sum of average job completion time and the average number of requested resources. The Markov decision process (MDP) problem was handled using the deep reinforcement learning strategy, which proposes the deep Q-network (DQN) algorithm, which keeps multiple responses in memory. It has the potential to increase the original DQN algorithm's performance. In addition, in a matching policy with limited time, the less demanded resource performs.

Xiaolan Liu et al., [23] proposed the resource allocation in Internet-of-Things (IoT) networks via edge computing using machine learning methodology. IoT network is a key component of edge computing. It is more convenient for users. 5064
To send the task to an edge server, the IoT network requires more resources. The jobs are carried out locally, requiring high processing capabilities and offloading strategies. It's already available technology that solves problems for IoT edge computing platforms. The newly designed optimal task offloading schemes for ultradense IoT edge computing networks in a statistics environment, which has the time-varying channel conditions, dynamic task queue, and computation capacity of the IoT users through machine learning approaches, were developed in a statistics environment. The two groups emerged from the computational offloading. The two categories are as follows:

i) Centralized user clustering

This scheme was utilized for the K-means clustering algorithm, and it is assigned the best and lowest user priority clustering, and it is implemented using edge computing and local computing.

ii) Distributed computation offloading scheme

Its long-term system cost is minimized using the DQN (deep Q-network) technique.

Dimitrios Dechouniotis et al., [24] stated that, the Internet of Things and the processing capability of local nodes were supplied for sensors, actuators for communications (IoT). The following factors are optimized, as are network, processing, and storage resources, as well as time feasibility and task assignment. To propose the DRUID-NET framework, this brings together Automata, Graph Theory, Machine Learning, Modern Control Theory, and Network Theory. The new approach addresses the foregoing issues by dynamically allocating resources. It covers resource analytic dynamical modeling, workload offered, networking environment, wireless communications, and mobile edge computing. The framework's major goal is to come up with new resource allocation strategies. It also has control over and decision-making approach. It offers features like service differentiation, context awareness, and ensuring QoS measurements.

Jiwei Huang et al., [25] discover the problem structure, establish a desirable quality for the TU coefficient matrix, and to introduce the -technique for converting LP problems. It's solved in an NP-hard way with the help of unimodular restrictions. Based on a real-life IoT dataset, the proposed mechanism inefficiently solved the task scheduling problem. Mobile edge computing has become more popular (MEC). Dynamic task scheduling and resource management are difficulties in this setting. Edge service providers' primary goal in the MEC environment is to maximize revenue. It, on the other hand, has used integer programming (IP)-based maximum task scheduling and resource management techniques.

III. A comparison of resource allocation technologies for edge computing

Table 1 summarizes the research findings in terms of algorithms/technology, parameters, simulation model, and benefits.

Algorithms/Technology	***Parameters***	***Simulation Model***	***Advantages***	***Disadvantages***
dynamic network resource demand predicting algorithm based on the group search optimizer (GSO) and incremental design of the RBF (GSO-INC-RBFDM).[1]	N/W cost, Weight, Pressure, Average Revenue, Accept rate	Mathematical Model	Improve the server solution; validate the user's dynamic resource requirements, To remove the maximum error value. Improve the performance for acceptance rate, network cost, link weight, and average revenue.	To increase resource utilization and network acceptance rate, it cannot function dynamic resource requirements and modify resource sharing conditions.
Blockchain technology and Deep learning approach, state-of-art machine learning algorithm, A3C (Advantage actor-critic), and combined AI and blockchains.[7]	Multiple service subscribers, QoS, Speed of network, and various tine delay channels constrain	Mathematical Model	To solve the multiple service subscriber resource allocation problem	The ECNs can offload the incoming data to the cloud if they don't have enough processing power to analyze it themselves, but doing so will result in longer reaction times and more network resource usage.
IoT-assisted edge computing, collaborative task scheduling shame.[8]	Time, throughput	Mathematical Model	an increased edge service throughput and guaranteeing deadlines of local IoT tasks	The Completion Time-Based Task Assignment method of estimating the remaining time is likely to be inaccurate if IoT devices perform irregular local activities after the estimation. However, the faults brought on by such activities do not accumulate over time because the most recent output production time precisely corresponds to the execution time of the earlier local tasks.
Machine-learning algorithm(Trustworthy community Detection Phase (TCDP), Matching Device Phase (MDP)) .[9]	Space, time size of devices	Mathematical Model	Reduces the complexity of service discovery task	Dynamic and static features are used to train machine learning models, but they also take into account static message sizes for each requester and static availability for each edge computer, such as the devices' continuous functioning. Analyzed the scenario in which these qualities are dynamic and change over time.
Caching and computation Model, heuristic algorithm.[10]	n/w cost, latency, n/w constraints, energy	Mathematical Model	Minimized energy server, reduced the complexity, modifies the cache placement, and improved task performance.	It won't work in a more broad example of non-uniform request distribution over various tasks/locations.
Independent Learners based Multi-Agent Q-learning (IL-based MA-Q) algorithm.[11]	Cost, power energy, radio access cost of IoT network	Algorithm	Avoided radio access collisions. Lower system cost, and improved for IoT Networks.	It cannot carry out actual experiments in a genuine setting to offer solutions to real-world issues.
RMF(Resource management Framework),Workload Balance Algorithm and Workload Re-distribution Strategy.[12]	Software company infrastructure, health checking, platform workload	Mathematical Model	systems supervision fail-rescue of software company infrastructure, health checking, and platform workload re-distribution	RMF integration with a commercial IoT platform is currently not possible.
Cooperative edge computing model, TCA-IPSO ALGORITHM.[13]	Time, an average of task	Mathematical Model	To minimize the average completion delay	The heuristic algorithm's complexity cannot be exactly calculated due to its inconsistency. To further increase the effectiveness of algorithm search, investigate new hybrid algorithms and parallel optimization techniques in the future.
Traffic-flow prediction algorithm(long short-term memory (LSTM) .[14]	N/w Traffic, latency packet loss	Mathematical Model	Reducing communication latency and decreasing the packet-loss ratio.	The mobile traffic-flow prediction and algorithm for dispatching communication resources, computing resources, together with the appropriate allocation scheme, and mobile traffic-flow control architecture based on an IoT-Cloud are not included in this research.
computation offloading	Time, latency	Mathematical Model	Reduce the average time for	-

framework, Lyapunov-based algorithm.[15]			probability	
an energy-efficiency-based resource allocation and power control algorithm.[16]	Power energy, time, bandwidth, subchannel, Quantity	Task Priority channel assignment algorithm (TP) , Uplink Equal Power channel allocation algorithm(UEP)	Increasing IoT terminals and average task offloading, optimal performance; reduced average energy consumption.	-
heuristic dynamic task processing algorithm, particle Swarm Optimization algorithm.[17]	Service Quality, System Power consumption, time delay. latency	iFogSim	to reduce the task latency time and system power consumption, improve the application service quality	The issue of where to place application modules and how to change the mapping of application modules and resource nodes dynamically during task execution to accommodate different IoT application services
Dynamic microservice scheduling scheme, the computational complexity of the scheduling algorithm. [18]	Network delay and Throughput, Energy rate, Failure Rate	Mathematical Model	improves the performance metrics for total network delay, average price, satisfaction level, energy consumption rate (ECR), failure rate, and network throughput	The adaptive microservice scheduling issue in edge-based healthcare systems
BID-PRoportional Allocation Mechanism (BID-PRAM), UNIform PRIcing Mechanism (UNI-PRIM), and FAIrness-seeking Differentiated PRIcing Mechanism (FAIDPRIM).[19]	Comparative analysis, time, Pricing	Mathematical Model	Identifies the advantages and disadvantages of various models, to give service providers guidance on various kinds of pricing schemes in edge computing.	-
multi-UE DNN framework, Iterative Alternating Optimization (IAO).[20]	Time, network factor, Latency, Bandwidth	Mathematical Theorem	The algorithms achieve minimizing the maximum delay among all UEs, system optimum with polynomial time complexity, and scales effectively. Also, validate the task of device-edge cooperative DNN partitioning archetype in the multi-user MEC environment	The difficult scenarios of shared computation, connectivity, and energy resource allocation for multi-user edge intelligence applications are taken into consideration in the work.
An optimization approach for computing resource allocation of huge IoHT technique. [21]	Network delay, security issues	MATLAB	the increased number of edge computing servers, the users, the computing capacity of devices, and task arrival rate; improve the performance of allocation strategy	-
deep Q-network (DQN) algorithm. [22]	completion time of jobs	(DQN) algorithm	Improve the performance of the original DQN algorithm. to minimize the long-term weighted sum of average completion time of jobs and an average number of requested resources	Massive channel access, defect detection, and other, less-studied aspects of quality of service
computational offloading Mechanism, *K*-means clustering algorithm, DQN (deep *Q*-network) algorithm. [23]	Size of job, Network factors, cost, Energy Factors, time	Algorithm	Highest and lowest user priority clustering, minimizing the long-term system cost.	-
DRUID-NET framework.[24]	QoS, Network metrics	Mathematical Model	It is controlling and decision strategy itself. Include service differentiation, context-awareness, guaranteeing the Quality of Service (QoS) metrics	-
TU coefficient matrix and introduce the λ-technique, Cross edge task scheduling algorithm. [25]	Network factors	Cross edge task scheduling algorithm	Achieved the task scheduling problem inefficiently manner based on a real-life IoT dataset.	The issue of multi-device task scheduling in edge computing from the standpoint of game theory, and attempt to uncover some game-theoretical features that aid in MEC system design.

IV.CONCLUSION

Edge computing favors time-critical applications that require increased accuracy, low latency, high-speed analytics, faster response time, improved reliability, and availability by collecting, processing, and analyzing data close to the data source before sending refined results to a centralized cloud. Combining edge and cloud computing could benefit IoT in a variety of ways. Edge computing is still in its development; hence thorough research into this new technology is required. This study looked at how computing paradigms are changing, as well as challenges and opportunities. This comprehensive analysis may aid prospective scholars in comprehending current advances in the study of evolving computer paradigms. This comprehensive survey might help future researchers understand recent breakthroughs in changing computer paradigms.

References

[1] Xiancui Xiao, Xiangwei Zheng, Tian Jie,"Dynamic resource allocation algorithm of virtual networks in edge computing networks", springer Personal and Ubiquitous Computing,20 July 2019.

[2] Garcia, P., Montresor, A., Epema, D., Datta, A., Higashino, T., Iamnitchi, A., Barcellos, M., Felber, P., Riviere, E.: Edge-centric computing: vision and challenges",ACM SIGCOMM Comput. Commun. Rev. 45(5), 37–42 (2015).

[3] Vaishali Sharma, Vipul Pratap Rai, Dr. K K Sharma3, "Edge Computing: Needs, Concerns and Challenges", International Journal of Scientific & Engineering Research, Volume 8, Issue 4, April-2017 ,ISSN 2229-5518.

[4] Auday Al-Dulaimy, Yogesh Sharma, Michel Gokan Khan, and Javid Taheri. Introduction to edge computing. In Edge Computing: Models, technologies and applications, pages 3-25, 2020.

[5] Ines Sitton-Candanedo(B), Ricardo S. Alonso, Sara Rodr ıguez-Gonzalez,Jos´e Alberto Garc´ıa Coria, and Fernando De La Prieta ",Edge Computing Architectures in Industry 4.0: A General Survey and Comparison", January 2020.

[6] Ines Sittón-Candanedo, Juan Manue Corchado," An Edge Computing Tutorial", Oriental Journal of Computer Science and Technology, ISSN: 0974-6471, Vol. 12, No. (2) 2019, Pg. 34-38.

[7] Ying He, Yuhang Wang, Chao Qiu, Qiuzhen Lin, Jianqiang Li, Zhong Ming, "Blockchain-based Edge Computing Resource Allocation in IoT: A Deep Reinforcement Learning Approach", IEEE Internet of Things Journal, Volume: 8, Issue: 4, Feb.15, 15 2021, Page(s): 2226 – 2237.

[8] Youngjin Kim, Chiwon Song1, Hyuck Han2, Hyungsoo Jung1,And Sooyong Kang, "Collaborative Task Scheduling for IoT-Assisted Edge Computing ",IEEE Access, Volume: 8, 02 December 2020, Page(s): 216593 - 216606.

[9] Abdullah Khanfor, Raby Hamadi, Hakim Ghazzai, Ye Yang, Mohammad Rafiqul Haider and Yehia Massoud, "Computational Resource Allocation for Edge Computing in Social Internet-of-Things", 2020 IEEE 63rd International Midwest Symposium on Circuits and Systems (MWSCAS),IEEE China Communications, Volume: 17, Issue: 9, Sept. 2020.

[10] Jiechen Chen, Hong Xing, Xiaohui Lin and Suzhi Bi,"Joint Cache Placement and Bandwidth Allocation for FDMA-based Mobile Edge Computing Systems",ICC 2020 - 2020 IEEE International Conference on Communications (ICC),7-11 June 2020.

[11] Xiaolan Liu; Jiadong Yu; Zhiyong Feng; Yue Gao, "Multi-agent reinforcement learning for resource allocation in IoT networks with edge computing", IEEE China Communications , Volume: 17, Issue: 9, Sept. 2020, Page(s): 220 – 236.

[12] Sasa Pesic; Milos Radovanovic; Mirjana Ivanovic, "An MQTT-based Resource Management Framework for Edge Computing Systems", IEEE,2020 International Conference on INnovations in Intelligent Systems and Applications (INISTA),24-26 Aug. 2020.

[13] Qianjun Wang; Sujie Shao; Shaoyong Guo; Xuesong Qiu; Zhili Wang,"Task Allocation Mechanism of Power Internet of Things Based on Cooperative Edge Computing", IEEE Access (Volume: 8) 31 August 2020,Page(s): 158488 - 158501.

[14] Min Chen; Yiming Miao; Hamid Gharavi; Long Hu; Iztok Humar, "Intelligent Traffic Adaptive Resource Allocation for Edge Computing-Based 5G Networks",IEEE Transactions on Cognitive Communications and Networking,Volume: 6, Issue: 2, June 2020,Page(s): 499 - 508.

[15] Jianhui Liu; Qi Zhang, "Computation Resource Allocation for Heterogeneous Time-Critical IoT Services in MEC",2020 IEEE Wireless Communications and Networking Conference (WCNC),25-28 May 2020.

[16] Zhang Bo-wei; Wu Wei-nong; Hu Xin; Xie Ying-zhao; Fu Quan-yong; Wang Jian; Li Jin-fu,"Joint Resource Allocation and Power Control Algorithm for Internet of Things based on Wireless Power Transfer and Edge Computing",2020 5th International Conference on Computer and Communication Systems (ICCCS),15-18 May 2020.

[17] Juan Fang; Aonan Ma, "IoT Application Modules Placement and Dynamic Task Processing in Edge-Cloud Computing",IEEE Internet of Things Journal (Early Access),07 July 2020 , Page(s): 1 – 1

[18] Amit Samanta and Jianhua Tang,"Dyme: Dynamic Microservice Schedulingin Edge Computing Enabled IoT", IEEE Internet of Things Journal,Volume: 7, Issue: 7, July 2020,Page(s): 6164 – 6174.

[19] Beomhan Baek; Joohyung Lee; Yuyang Peng; Sangdon Park,"Three Dynamic Pricing Schemes for Resource Allocation of Edge Computing for IoT Environment",IEEE Internet of Things Journal , Volume: 7, Issue: 5, May 2020,Page(s): 4292 - 4303.

[20] Xin Tang, Xu Chen, Liekang Zeng, Shuai Yu, and Lin Chen, "Joint Multi-User DNN Partitioning and Computational Resource Allocation for Collaborative Edge Intelligence",IEEE Internet of Things Journal,Volume: 8, Issue: 12, June15, 15 2021,Page(s): 9511 - 9522.

[21] Jianxi Wang, Liutao Wang, "A Computing Resource Allocation Optimization Strategy for Massive Internet of Health Things Devices Considering Privacy Protection in Cloud Edge Computing Environment", Springer Journal of Grid Computing, Issue 2/2021, volume 19, 03 April 2021.

[22] Xiong Xiong, Kan Zheng, Lei Lei , and Lu Hou, "Resource Allocation Based on Deep Reinforcement Learning in IoT Edge Computing", IEEE Journal on Selected Areas in Communications, Volume: 38, Issue: 6, June 2020,Page(s): 1133 - 1146.

[23] Xiaolan Liu; Jiadong Yu; Jian Wang; Yue Gao,"Resource Allocation With Edge Computing in IoT Networks via Machine Learning", IEEE Internet of Things Journal, Volume: 7, Issue: 4, April 2020,Page(s): 3415 - 3426.

[24] Dimitrios Dechouniotis, Nikolaos Athanasopoulos, Aris Leivadeas, Nathalie Mitton, Raphaël M. Jungers, Symeon Papavassiliou:"Edge Computing Resource Allocation for Dynamic Networks: The DRUID-NET Vision and Perspective." Sensors 20(8): 2191, 14 April 2020.

[25] Jiwei Huang, Songyuan Li, Ying Chen, "Revenue-optimal task scheduling and resource management for IoT batch jobs inmobile edge computing," Springer Peer-to-Peer Networking and Applications, 06 march 2020.