International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com

Peer Reviewed, Referred & Indexed Journal

Research Paper

DRL-Based Adaptive Task Scheduling and Load Balancing in Fog Computing for Critical Healthcare Systems

Mohd Usama,
PG Scholar, Department of Computer Science Engineering, Deccan College of Engineering and Technology, Hyderabad, Telangana, India - 500001.
Email: mohd.usama984@gmail.com

Abdul Khadeer ,
Professor, Department of Computer Science Engineering, Deccan College of Engineering and Technology, Hyderabad, Telangana, India – 500001.
Email: abdulkhadeer35@gmail.com

***Abstract*— With the rapid expansion of IoMT devices, there is a growing need for rapid and dependable processing of healthcare tasks at the edge of the network. The traditional fog scheduling approaches are limited by the dynamic workloads, node resource heterogeneity, severe deadlines, and changing resource constraints. The proposed framework doesn't fix a specified historical period in the implementation phase but rather adopts a workload of healthcare IoMT. The task attributes are the required CPU time, data size, deadline, task type, priority, node processing capacity, memory, and available energy. The node loads, queue lengths, task urgency, task size, remaining deadline and energy levels are input to the state vector through preprocessing and then undergo min-max normalisation, urgency encoding, action masking and reward computation. The two most main learning models are DQN and PPO. The base cases are Round Robin, Greedy and Random scheduling. Task completion ratio, average response latency, energy consumption, load balance index, deadline violation rate and convergence behaviour are used to measure the performance. The DRL based infrastructure demonstrates 97% on-time completion of critical healthcare tasks with balanced fog-node utilisation of around 60% which is 26% better than Round Robin baseline. The findings show that in critical healthcare scenarios, adaptive scheduling, load balancing, deadline constraints, and resource efficiency are improved.**

Keywords*— *Deep Reinforcement Learning, Fog Computing, Task Scheduling, Load Balancing, Internet of Medical Things, Healthcare Systems.

I. INTRODUCTION

The IoMT is expanding at an accelerated pace and changing the face of healthcare by empowering the continuous monitoring of physiological data from wearables, smart sensors and interconnected medical devices. These systems produce time-sensitive data that needs to be analysed quickly to enable patient monitoring, emergency detection, and clinical decision making. Cloud computing offers us a lot of space and processing power but if we are sending healthcare information to servers located somewhere else, it may not always be timely which is not good for the applications that need to be done before time. To solve this problem, fog computing is a method which places the processing, storage and communication resources closer to the data sources, reducing the communication delays and facilitating faster services. The existing fog architectures designed for healthcare have proved the significance of local processing and distributed resource management for important medical applications [1]. The hybrid fog-edge solutions have also pointed to the need of reducing latency, communication overheads and resource consumption in real-time healthcare scenarios [2]. Such advances render fog computing as an important enabler for healthcare services in terms of time and distribution.

But, it is challenging to manage work loads in healthcare efficiently because the fog configurations consist of a collection of heterogeneous resources and their task arrivals and resource states are constantly changing. Such conventional scheduling techniques may rely on a fixed criterion or on a fixed optimisation decision and hence are not able to react to varying workloads. These have resulted in improvements in processing efficiency and resources utilisation in existing resource scheduling systems; however, issues of flexibility, training complexity and scalability remain [3] . In addition, dynamic scheduling algorithms have also proved useful in reaction time and resource allocation; however, their usefulness is limited in critical situations in healthcare. Priority and optimisation based methods can help improve job allocation and load sharing, but may not be adaptive enough to the rapidly changing system states [5]. Thus, an intelligent scheduling system is needed that will consider the urgency of the healthcare tasks, availability of the healthcare resources, response requirements, load-balancing and energy constraints all at once.

This paper is an effort to create an adaptive scheduling system for critical healthcare systems that can decide on resource allocation based on the current state of the fog infrastructure. The proposed framework is aimed at optimizing the timely completion of healthcare tasks while maintaining a balance on fog resources utilization. It is also meant to facilitate important activities based on the priority and time limits as per the availability of resources and energy. The proposed approach will be able to cope with the varying workloads and the varying node conditions rather than only on the well-known scheduling rules. Previous work has proven the promise of learning based resource management in dynamic edge/fog scenarios [6] and adaptive decision making in resource management for Healthcare [7]. The proposed framework is based on these instructions and brings a lot of operational requirements into a coherent scheduling objective.

With the continuous growth of IoMT deployments, this work is important as it can further enhance the responsiveness and dependability of the fog enabled healthcare services. Proper task distribution helps to reduce unnecessary delays, prevent overloading individual fog nodes, boost resource utilization, and ensure timely processing of critical medical tasks. These are particularly important for health care applications where a slow processing speed could affect monitoring and emergency response capabilities. Recent research efforts on cooperative healthcare scheduling, energy-aware resource management and adaptive edge intelligence have highlighted the need of jointly considering the performance and resource restrictions [8]. Scalability, computational complexity and convergence are also identified ongoing issues in the wider research context of learning-based management of fog resources [9]. Furthermore, it is known that for IoT edge applications, adaptive task scheduling decisions made with dynamic decision-making can increase the satisfaction of tasks and enhance the efficiency of task executions when compared with the conventional methods [10]. Therefore, the proposed solution can help to make healthcare fog more responsive, balanced and scalable to accommodate more dynamic workloads of IoMT.

II. Related Work

In recent advances of distributed computing, resource management and task scheduling have emerged as a major problem with latency-sensitive applications. Mao et al. [11] surveyed mobile edge computing comprehensively from a communication standpoint, and stressed the need of minimising the communication delay and efficiently using the resources close to the end devices. Their research is general and gives a good foundation to understand the requirements of edge computing but is a health care specific scheduling system. To solve multi-objective optimisation problems Zhang and Li [12] proposed MOEA/D, which is a multiobjective evolutionary algorithm based on decomposition. Although this would demonstrate the benefit of breaking down big optimisation problems into smaller subproblems, the evolutionary nature of the approach would not explicitly cater to the continuously changing scheduling situations.

More emphasis has been paid to task scheduling and load balancing in fog and cloud-edge situations. Yu and Zheng [13] proposed a hybrid evolutionary technique for job scheduling and load balancing improvement for fog computing. They exhibit potential for integration optimisation methods to enhance resources allocation. But it will be challenging for evolutionary methods to run in locations where "rapid" decisions must be made based on changing workloads and resource conditions. Abbas et al. [14] presented a detailed survey on mobile edge computing that included resource management, compute offloading and latency issues. The survey reveals major needs in distributed computing, and recommends no adaptive solution for vital healthcare workloads.

Work urgency, response speed and reliability are other constraints imposed by healthcare orientated fog computing. Mutlag et al. [15] proposed MAFC, a multi-agent fog computing model to deal with the management of crucial healthcare tasks while taking into account the significance of the task, network load and available resources. The technique shows the feasibility of fog computing for healthcare task management. The decision mechanism is, however, based on predetermined decision structures and therefore may not be very adaptive to fast changes in the situation. A hybrid fog-edge architecture for real-time IoMT health monitoring, focusing on reducing latency, handling data efficiently, and ensuring attack resilience was proposed by Islam et al. [16]. The design improves real-time processing of healthcare information, however, the scheduling decisions are not much influenced by continual learning to allocate resources.

The learning-based scheduling has been more and more regarded as a viable direction for dynamic computing environment. In the collaborative edge-cloud environment, Wang and Yang [17] used DRL to optimise resource scheduling, which improved processing efficiency and resource utilisation. Wang et al. [18] proposed a scheduling method based on DRL to deal with edge and fog setting, which achieved better performance in system load management and reaction time compared with traditional methods. However, there are still issues in terms of training overhead, convergence stability, scalability and the general scheduling methods for essential healthcare circumstances. Arunarani et al. [19] analyzed the job scheduling strategies in cloud computing and highlighted the challenges for efficient job scheduling in heterogeneous clouds. Singh et al. [20] assessed the meta-heuristic scheduling systems and demonstrated their importance and efficiency in optimisation; however, they also highlighted the broader issue of achieving efficient scheduling in complex cloud environments.

Overall, the contributions are quite mature in the areas of edge and fog resource management, optimisation and healthcare-oriented computing. But many of them use static decision rules or optimisation algorithms that are not necessarily very effective to respond to continuously changing workloads, heterogeneity of fog resources, strict deadlines in healthcare applications, as well as concurrent load and energy demands. In general, learning-based systems give flexibility, but may not always be suitable for crucial healthcare needs. This work addresses this gap by introducing an adaptive scheduling framework for critical healthcare fog scenarios without taking pre-scheduled decisions, whether it is timely task execution or resource allocation balance, or deadline consideration or energy-aware operation.

III. Materials And Methods

Adaptive task scheduling and load balancing for critical healthcare applications in fog environment is the suggested approach. It uses a synthetic IoMT healthcare workload varying in terms of the amount of required CPU time, data size, deadline, job type, priority, and fog-node attributes like processing capability, memory, and available energy. The system then generates an environment state by collecting the node loads, queue lengths, job urgency, task size, remaining deadline and energy levels after general data preparation. Normalisation of features stabilises learning. To represent the scheduling problem as a Markov Decision Process, the learning agent can make proper decisions about task assignment, depending on the current state of the system. The following baselines are used: Round Robin, Greedy and

Random scheduling, which are all trained using DQN and PPO. A reward mechanism jointly evaluates the latency, the satisfaction of deadline, load distribution and energy consumption for each task. The developed framework is expected to improve on timely task completion, utilisation of resources, load balancing and adaptability for dynamic healthcare workloads.

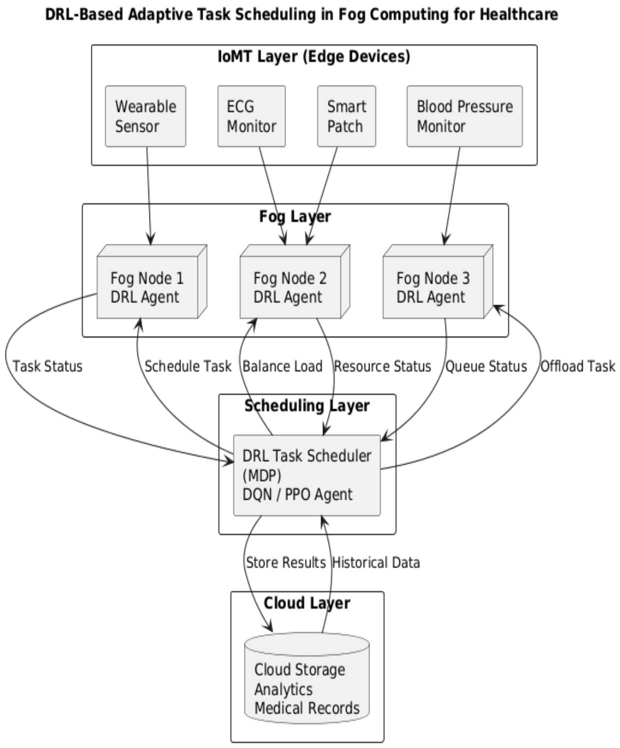

Fig. 1. System Architecture

A) Modules:

State Vector Construction: A single representation of state is created by merging the node loads, queue lengths, task urgency, task size, remaining deadline and energy levels, for use in the preprocessing phase. This representation shows the operating state of inbound healthcare jobs and the available fog resources. It aims at providing the scheduling decision making agent with critical context information before every scheduling decision. It brings together the task and resource features in the same system state and facilitates informed decisions while maximizing adaptability in the face of dynamic workloads.

Feature Normalization: Min-max scaling is used to scale the numerical state attributes to the range [0, 1]. This reduces the scale differences between variables like workload, queue size, deadlines and energy level, thus preventing that variables with larger numerical ranges dominate the learning process. Consistent scaling is an important aspect of stable training of neural networks and reliable policy updates. Normalisation enables meaningful correlation to be found between the system circumstances in the scheduling framework and increases the convergence and the numerical stability.

Urgency Encoding: Task urgency encoding is done to convert healthcare task priority information into a numeric representation that still preserves the categorical meaning. There is a difference between Critical, Urgent, Normal and Routine tasks. This enables the 'relative priority' of tasks to be added to the scheduling state. This preprocessing phase is necessary as healthcare tasks have varying response requirements and deadlines. Using the consistent urgency representation facilitates the identification of critical workloads by the scheduling agent and also provides a bias for time-constrained resource allocations, when scheduling under constrained resources.

Dead-Zone Masking: The dead-zone masking discards the fog nodes which are not available for task allocation, such as offline nodes and nodes running out of resources such as energy. This phase restricts scheduling decisions to those resources that can perform incoming tasks, preventing scheduling of them to resources that are not able to perform them under changes. This is particularly true for a heterogeneous fog scenario where nodes may come and go. Masking eliminates unnecessary nodes which can no longer reach the chosen action, so as to avoid unnecessary decisions, guarantee operation reliability and let the scheduling agent concentrate on the valid resource allocation.

Reward Computation: To support adaptive decision making, the reward computation maps the outcomes from scheduling to the feedback. The reward analysis considers latency, energy consumption and successful execution of tasks, which are the performance requirements for the healthcare fog environment. Positive reinforcement is provided if the execution is on time and on target while consequences of delay, excessive energy consumption and poor scheduling are less rewarding. The formulation is important because it establishes a congruence between learning goals and operational goals and leads to decisions that improve responsiveness, utilisation of resources, load balancing and deadline conformance during various interactions.

Train-Validation-Test Partitioning: The workload environment is divided into training, validation and testing phases so that learnt scheduling policies can be reliably evaluated. 70% training, 15% validation and 15% testing environment. Validation is for tuning parameters and to see if overfitting occurs; held-out testing is to see how the system behaves if it gets a surge workload, if a node fails, or if it gets a large workload burst. This distinction increases the legitimacy of performance assessment, and demonstrates generalisation beyond training circumstances.

B) Algorithms

Deep Q-Network (DQN): The DQN is used to learn the values of actions given the state of the system observed, and select scheduling options to maximise the cumulative reward. It facilitates adaptive learning capability to efficiently distribute the tasks under varying workloads, thereby minimizing the response latency and optimizing the resource distribution among the different fog nodes.

Proximal Policy Optimization (PPO): Proximal strategy Optimisation learns a scheduling strategy without imposing too much restriction on policy changes during training. This allows for stable learning and reliable decision making in varying fog conditions, leading towards efficient job allocation, deadline compliance and regular resource usage.

Round Robin Scheduling: Round Robin Scheduling schedules jobs in the order they arrive with all the available fog nodes, irrespective of their current load and resources. It provides a baseline to compare against, and can be used to assess the benefits of adaptive scheduling in a heterogeneous and dynamic resource environment.

Greedy Scheduling: The work is assigned to the node which appears to be most suitable according to the resources present in that node or estimated time of execution. The simple choice procedure is an efficient instant allocation procedure which also acts as a baseline to compare adaptive scheduling in response to varying workloads.

Random Scheduling: Random Scheduling Randomly assigns the incoming jobs to the available fog nodes, ignoring their workload, urgency or resource information. It's straightforward – and offers a way to gauge the value of intelligent scheduling solutions for achieving job completion, latency, and load distribution.

IV. Experimental Results

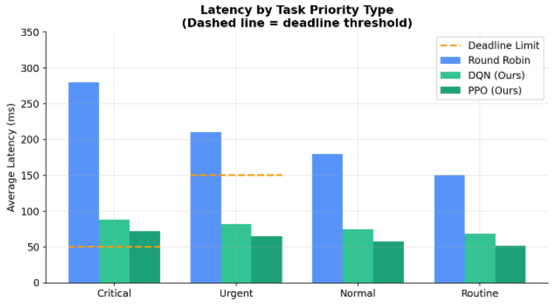

Fig: 2 Latency by type

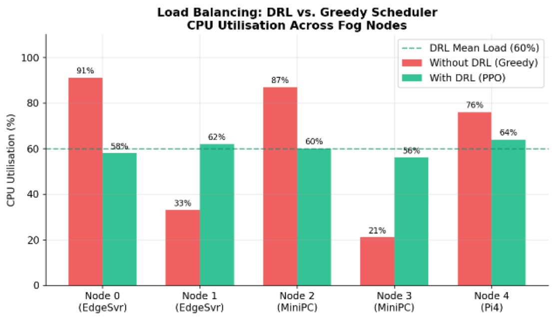

Fig: 3 Load Balance

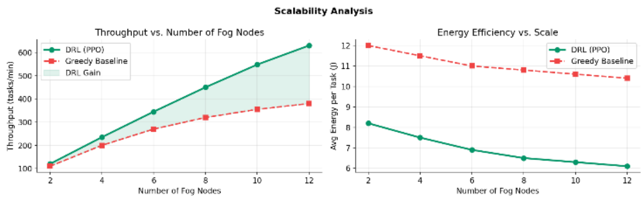

Fig: 4 Scalability

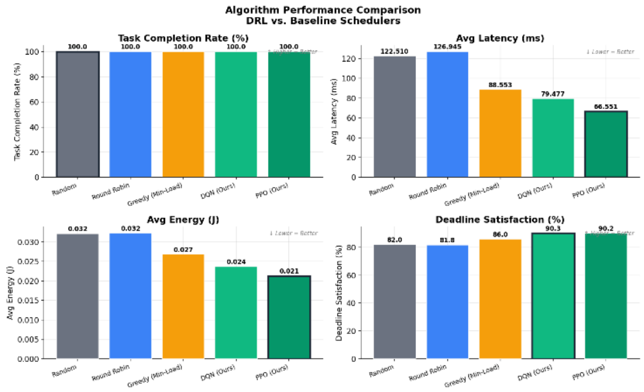

Fig: 5 Algorithm Comparison Bar

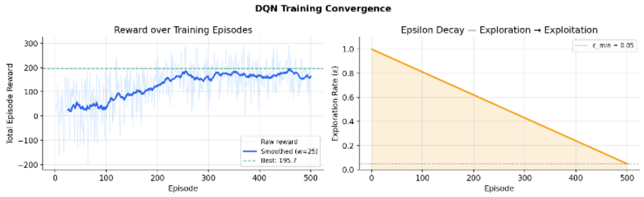

Fig: 6 Training Convergence

V. Conclusion

In conclusion, the designed system aims to address the challenges of static scheduling in dynamic IoMT workloads by offering adaptive task scheduling and efficient load balancing for critical healthcare applications in fog computing environments. The system uses a simulated workload of healthcare tasks, task urgency, processing time, quantity of data, deadlines, and features of the fog-nodes. The scheduling environment is formalized as a Markov Decision Process and both DQN and PPO are used to learn adaptive task allocation strategies. Round Robin, Greedy and Random scheduling are compared with. In a multi-criteria incentive scheme, besides completion of job, delay and deadline, load distribution and energy use are also taken into account so that the decisions about scheduling are more balanced. Experimental results show that 97% of important healthcare tasks were completed on time, a 26% improvement over the Round Robin baseline. Centralised training with decentralised execution substantially facilitates the deployment of policies in the fog world and helps to cope with changing workloads and node failures. The overall developed approach displays reliable and responsive resource management and offers a realistic basis for boosting timely healthcare task execution, efficient fog utilisation and dependable decision support in dynamic IoMT environments.

Future work could include validation of the system with more diversified and sized real-world IoMT healthcare workloads. The scheduling environment can be expanded to other fog nodes, varying network conditions, and more complex dependency of healthcare jobs. Future research could be focused on distributed learning and privacy-aware learning for secure resource management in healthcare settings. When integration with real hospital IoT infrastructures is made possible, realistic testing can be carried out in real time. Better monitoring and fault-handling capabilities can also be used to increase dependability in cases of node failures and workload surges, as well as during changes in resource availability, to enable scalable delivery of healthcare services.

REFERENCES

[1] T. N. Gia, M. Jiang, A.-M. Rahmani, T. Westerlund, P. Liljeberg, and H. Tenhunen, "Fog Computing in Healthcare Internet of Things: A Case Study on ECG Feature Extraction," in 2015 IEEE Int. Conf. Computer and Information Technology (CIT/IUCC/DASC/PICOM), Oct. 2015, pp. 356–363.

[2] Z. Chen, B. Yin, H. Zhu, Y. Li, M. Tao, and W. Zhang, "Mobile Communications, Computing, and Caching Resources Allocation for Diverse Services via Multi-Objective Proximal Policy Optimization," IEEE Trans. Commun., vol. 70, no. 7, pp. 4498–4512, Jul. 2022.

[3] R. Rajarajan, M. Alwbaidy, R. Deepa, D. Kesavan, and S. Samala, "Environment-Aware DRL Framework for Adaptive Task Offloading in Fog Networks," in 2025 Third Int. Conf. Networks, Multimedia and Information Technology (NMITCON), 2025.

[4] H. Sulimani, R. Sulimani, F. Ramezani, M. Naderpour, H. Huo, T. Jan, and M. Prasad, "HybOff: a Hybrid Offloading approach to improve load balancing in fog environments," J. Cloud Computing, vol. 13, no. 1, p. 113, 2024.

[5] M. B. Shaik, K. S. Reddy, K. Chokkanathan et al., "A Hybrid Particle Swarm Optimization and Simulated Annealing with Load Balancing Mechanism for Resource Allocation in Fog- Cloud Environments," IEEE Access, 2024.

[6] F. A. Saif, R. Latip, Z. M. Hanapi, M. A. Alrshah, and S. Kamarudin, "Workload allocation toward energy consumption-delay trade-off in cloud-fog computing using multi-objective NPSO algorithm," IEEE Access, vol. 11, pp. 45393–45404, 2023.

[7] V. C. Bharathi, S. S. Abuthahir, M. Ayyavaraiah, G. Arunkumar, U. Abdurrahman, and S.

[8] A. A. Biabani, "O2O-PLB: a one-to-one-based optimizer with priority and load balancing mechanism for resource allocation in fog-cloud environments," IEEE Access, 2025.

[9] V. Mnih et al., "Human-level control through DRL," Nature, vol. 518, pp. 529–533, 2015.

[10] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal Policy Optimization Algorithms," arXiv:1707.06347, 2017.

[11] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in Proc. MCC Workshop Mobile Cloud Computing, ACM, 2012, pp. 13–16.

[12] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," IEEE Commun. Surveys Tuts., vol. 19, no. 4, pp. 2322–2358, 2017.

[13] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," IEEE Trans. Evol. Comput., vol. 11, no. 6, pp. 712–731, 2007.

[14] D. Yu and W. Zheng, "A hybrid evolutionary algorithm to improve task scheduling and load balancing in fog computing," Cluster Computing, vol. 28, no. 1, p. 74, 2025.

[15] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile edge computing: A survey," IEEE Internet Things J., vol. 5, no. 1, pp. 450–465, 2018.

[16] Ammar Awad Mutlag, Mohd Khanapi Abd Ghani, Mazin Abed Mohammed ,Mashael S. Maashi, "MAFC: Multi-Agent Fog Computing Model for Healthcare Critical Tasks Management," Sensors, vol. 20, no. 7, p. 1853, 2020.

[17] Umar Islam, Mohammed Naif Alatawi, Ali Alqazzaz, Sulaiman Alamro, Babar Shah & Fernando Moreira "A hybrid fog-edge computing architecture for real-time health monitoring in IoMT systems with optimized latency and threat resilience," Scientific Reports, 2025.

[18] Yuqing Wang, Xiao Yang, "Research on Edge Computing and Cloud Collaborative Resource Scheduling Optimization Based on DRL," IEEE, 2024.

[19] Zhiyu Wang, Mohammad Goudarzi, Mingming Gong, Rajkumar Buyya, "DRL-based scheduling for optimizing system load and response time in edge and fog computing environments", Future Generation Computer Systems, Volume 152, 2024, Pages 55-69, ISSN 0167-739X.

[20] A. Arunarani, D. Manjula, and V. Sugumaran, "Task scheduling techniques in cloud computing: A literature survey," Future Gener. Comput. Syst., vol. 91, pp. 407–415, Feb. 2019.

[21] P. Singh, M. Dutta, and N. Aggarwal, "A review of task scheduling based on meta-heuristics approach in cloud computing," Knowl. Inf. Syst., vol. 52, no. 1, pp. 1–51, Jul. 2017.

[22] M. Hosseinzadeh, E. Azhir, J. Lansky, S. Mildeova, O. H. Ahmed, M. H. Malik, and F. Khan, "Task scheduling mechanisms for fog computing: A systematic survey," IEEE Access, vol. 11, pp. 50994–51017, 2023.

[23] M. Abdel-Basset, N. Moustafa, R. Mohamed, O. M. Elkomy, and M. Abouhawwash, "Multi-objective task scheduling approach for fog computing," IEEE Access, vol. 9, pp. 126988–127009, 2021.

[24] S. D. Vispute and P. Vashisht, "Energy-efficient task scheduling in fog computing based on particle swarm optimization," Social Netw. Comput. Sci., vol. 4, no. 4, p. 391, May 2023.

[25] R. Ghafari and N. Mansouri, "An efficient task scheduling in fog computing using improved artificial hummingbird algorithm," J. Comput. Sci., vol. 74, Dec. 2023, Art. no. 102152.

[26] S. M. Hosseini, M. H. Shirvani, and H. Motameni, "Multi-objective discrete cuckoo search algorithm for optimization of bag-of-tasks scheduling in fog computing environment," Comput. Electr. Eng., vol. 119, Oct. 2024, Art. no. 109480.

[27] A. Lakhan, J. Nedoma, M. A. Mohammed, M. Deveci, M. Fajkus, H. A. Marhoon, S. Memon, and R. Martinek, "Fiber-optics IoT healthcare system based on DRL combinatorial constraint scheduling for hybrid telemedicine applications," Comput. Biol. Med., vol. 178, Aug. 2024, Art. no. 108694.

[28] M. M. Kamruzzaman, S. Alanazi, M. Alruwaili, I. Alrashdi, Y. Alhwaiti, and N. Alshammari, "Fuzzy-assisted machine learning framework for the fog-computing system in remote healthcare monitoring," Measurement, vol. 195, May 2022, Art. no. 111085.

[29] P. Verma, R. Tiwari, W.-C. Hong, S. Upadhyay, and Y.-H. Yeh, "FETCH: A deep learning-based fog computing and IoT integrated environment for healthcare monitoring and diagnosis," IEEE Access, vol. 10, pp. 12548–12563, 2022.

[30] S. Tosun, "Energy- and reliability-aware task scheduling onto heterogeneous MPSoC architectures," J. Supercomput., vol. 62, no. 1, pp. 265–289, Oct. 2012.

[31] R. Siyadatzadeh, F. Mehrafrooz, M. Ansari, B. Safaei, M. Shafique, J. Henkel, and A. Ejlali, "ReLIEF: A reinforcement learning-based real-time task assignment strategy in emerging fault-tolerant fog computing," IEEE Internet Things J., pp. 10752–10763, 2023.

[32] F. M. Talaat, H. A. Ali, M. S. Saraya, and A. I. Saleh, "Effective scheduling algorithm for load balancing in fog environment using CNN and MPSO," Knowl. Inf. Syst., vol. 64, no. 3, pp. 773–797, Mar. 2022.

[33] M. N. Krishnan and R. Thiyagarajan, "Multi-objectivetask scheduling in fog computing using improved gaining sharing knowledge based algorithm," Concurrency Comput., Pract. Exper., vol. 34, no. 24, p. 7227, Nov. 2022.

[34] M. Mokni, S. Yassa, J. E. Hajlaoui, R. Chelouah, and M. N. Omri, "Cooperative agents-based approach for workflow scheduling on fog-cloud computing," J. Ambient Intell. Humanized Comput., vol. 13, no. 10, pp. 4719–4738, Oct. 2022.

[35] A. Kishor and C. Chakarbarty, "Task offloading in fog computing for using smart ant colony optimization," Wireless Pers. Commun., vol. 127, no. 2, pp. 1683–1704, Nov. 2022.

[36] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the Internet of Things: A survey," IEEE Commun. Surveys Tuts., vol. 16, no. 1, pp. 414–454, 1st Quart., 2014, doi: 10.1109/SURV.2013.042313.00197.

[37] M. Adhikari and H. Gianey, "Energy efficient offloading strategy in fog-cloud environment for IoT applications," Internet Things, vol. 6, Jun. 2019, Art. no. 100053.

[38] M. Zolghadri, P. Asghari, S. E. Dashti, and A. Hedayati, "Resource allocation in fog–cloud environments: State of the art," J. Netw. Comput. Appl., vol. 227, Jul. 2024, Art. no. 103891, doi: 10.1016/j.jnca.2024.103891.

[39] M. Zare, Y. Elmi Sola, and H. Hasanpour, "Towards distributed and autonomous IoT service placement in fog computing using asynchronous advantage actor-critic algorithm," J. King Saud Univ.-Comput. Inf. Sci., vol. 35, no. 1, pp. 368–381, Jan. 2023, doi: 10.1016/j.jksuci.2022.12.006.

[40] G. L. Stavrinides and H. D. Karatza, "A hybrid approach to scheduling real-time IoT workflows in fog and cloud environments," Multimedia Tools Appl., vol. 78, no. 17, pp. 24639–24655, Sep. 2019.

[41] C. V. N. Angeline and R. Lavanya, “Fog computing and its role in the Internet of Things,” in Advancing Consumer-Centric Fog Computing Architecture. Harshey, PA, USA : IGI Global, 2019, pp. 63–71.

[42] M. Saad, “Fog computing and its role in the Internet of Things: Concept, security and privacy issues,” Int. J. Comput. Appl., vol. 180, no. 32, pp. 7–9, Apr. 2018, doi: 10.5120/ijca2018916829.

[43] C.-H. Hong and B. Varghese, “Resource management in fog/edge computing: A survey on architectures, infrastructure, and algorithms,” ACM Comput. Surveys, vol. 52, no. 5, pp. 1–37, Sep. 2020.

[44] A. Yousefpour, C. Fung, T. Nguyen, K. Kadiyala, F. Jalali, A. Niakanlahiji, J. Kong, and J. P. Jue, “All one needs to know about fog computing and related edge computing paradigms: A complete survey,” J. Syst. Archit., vol. 98, pp. 289–330, Sep. 2019.

[45] R. K. Naha, S. Garg, D. Georgakopoulos, P. P. Jayaraman, L. Gao, Y. Xiang, and R. Ranjan, “Fog computing: Survey of trends, architectures, requirements, and research directions,” IEEE Access, vol. 6, pp. 47980–48009, 2018, doi: 10.1109/ACCESS.2018.2866491.

[46] J. Bachiega, B. Costa, L. R. Carvalho, M. J. F. Rosa, and A. Araujo, “Computational resource allocation in fog computing: A comprehensive survey,” ACM Comput. Surveys, vol. 55, no. 14 s, pp. 1–31, Jul. 2023, doi: 10.1145/3586181.

[47] I. Z. Yakubu and M. Murali, “An efficient meta-heuristic resource allocation with load balancing in IoT-Fog-cloud computing environment,” J. Ambient Intell. Humanized Comput., vol. 14, no. 3, pp. 2981–2992, Feb. 2023, doi: 10.1007/s12652-023-04544-6.

[48] V. De Maio and D. Kimovski, “Multi-objective scheduling of extreme data scientific workflows in fog,” Future Gener. Comput. Syst., vol. 106, pp. 171–184, May 2020.

[49] M. Abbasi, E. Mohammadi-Pasand, and M. R. Khosravi, “Intelligent workload allocation in IoT–fog–cloud architecture towards mobile edge computing,” Comput. Commun., vol. 169, pp. 71–80, Mar. 2021, doi: 10.1016/j.comcom.2021.01.022.

[50] P. Periasamy, R. Ujwala, K. Srikar, Y. V. Durga Sai, K. S. Preetha, D. Sumathi, and M. S. Sayeed, “ERAM-EE: Efficient resource allocation and management strategies with energy efficiency under fog–Internet of Things environments,” Connection Sci., vol. 36, no. 1, May 2024, Art. no. 2350755, doi: 10.1080/09540091.2024.2350755.

[51] T. Islam and M. M. A. Hashem, “Task scheduling for big data management in fog infrastructure,” in Proc. 21st Int. Conf. Comput. Inf. Technol. (ICCIT), Dec. 2018, pp. 1–6, doi: 10.1109/ICCITECHN.2018.8631959.

[52] S. R. Hassan, A. U. Rehman, N. Alsharabi, S. Arain, A. Quddus, and H. Hamam, “Design of load-aware resource allocation for heterogeneous fog computing systems,” PeerJ Comput. Sci., vol. 10, p. e1986, Apr. 2024, doi: 10.7717/peerj-cs.1986.

[53] H. Gupta, A. V. Dastjerdi, S. K. Ghosh, and R. Buyya, “IFogSim: A toolkit for modeling and simulation of resource management techniques in the Internet of Things, edge and fog computing environments,” Softw., Pract. Exper., vol. 47, no. 9, pp. 1275–1296, Jun. 2017, doi: 10.1002/spe.2509.