Journal of Harbin Engineering University
ISSN: 1006-7043

Vol 45 No. 2
February 2024

Enabling Intelligent Mobility: A Survey of Vehicular Edge Computing and Task Offloading

T.Sivaprakash[1] , Dr.R.Sarala[2]
[1] Research Scholar, Department of Computer Science and Engineering,
Puducherry Technological University
[2] Professor, Department of Computer Science and Engineering,
Puducherry Technological University

Abstract
A new technology called vehicular edge computing (VEC) has the potential to revolutionize vehicular networks by harnessing the compute power of edge servers and nearby vehicles. Task offloading is a crucial mechanism within VEC for transferring computational tasks from the onboard resources of the vehicles to edge servers or peer vehicles. A number of task offloading strategies are discussed and categorized in this paper, from local, edge, and V2V offloading to collaborative, V2I, and V2V offloading. Following that, this study examines a variety of offloading techniques, including machine learning-based offloading and optimization algorithms. Besides mobility dynamics, resource management, security, privacy, and scalability considerations, the survey also focuses on the multifaceted challenges associated with VEC. The challenges are analyzed in relation to dynamic vehicular environments. The report also highlights emerging trends in VEC driven by technologies including 5G, IoT, and AI. Researchers, practitioners, and policymakers can gain insight and guidance from this survey on task offloading in virtual environments. Aside from capturing current research, it illuminates future directions, highlighting task offloading's profound impact on the evolution of vehicular edge computing.

Keywords: VEC, offloading, edge computing, Vehicular networks

1. Introduction

During the last few years, mobile and vehicular computing has undergone a transformation resulting from the convergence of vehicular networks and edge computing. In vehicular networks, Vehicle Edge Computing (VEC), an emerging paradigm, uses the computational capabilities of cars and nearby edge servers to address the growing demands for data-intensive applications, real-time decision-making, and low-latency services. This paradigm relies on the central concept of "task offloading."A fundamental mechanism in VEC is task offloading, which transfers computation tasks and data from a vehicle's onboard resources to edge servers or other vehicles. In this way, edge computing enhances efficiency, reliability, and scalability of vehicular services by enabling collaboration, sharing of resources, and collectively harnessing the power of edge computing.

Task offloading has a profound impact on the advancement of VEC and has the potential to revolutionize a wide variety of vehicular applications, which motivates this comprehensive survey paper. With the advent of increasingly sophisticated sensors, cameras, and computing power in vehicles, a vast amount of data is generated, and demands instantaneous processing for applications from autonomous driving to traffic management to in-vehicle entertainment systems [1].

A number of emerging technologies, including 5G and beyond, the Internet of Things (IoT), and artificial intelligence, have intensified the need for efficient and intelligent task offloading strategies [2].Offloading tasks in Vehicular Edge Computing (VEC) is an important component of VEC because of its multiple implications that improve network efficiency and functionality. Through task offloading, computational latency is reduced for safety-critical applications, enabling real-time responses [3]. In addition, it optimizes resource utilization by intelligently distributing computational tasks, resulting in a reduction in energy consumption and an extended lifespan for onboard devices [4]. Furthermore, task offloading helps VEC services scale, allowing them to meet the growing demand for dynamic vehicular networks [5]. In addition, it protects data privacy and security by limiting sensitive data processing within vehicles [6]. By alleviating network burdens and reducing contention for limited network resources, task offloading enhances the overall performance of VEC systems [7]. Task offloading's diverse benefits underscore how important it is for VEC networks to survive and thrive in the future.

Taking these reasons into account, this survey paper aims to provide a comprehensive overview of task-offloading research in the context of VEC. Our study explores the various types of offloading, the techniques employed, the challenges faced, and the emerging trends that shape VEC's future. This paper provides valuable guidance for researchers, practitioners, and policymakers interested in the evolving field of vehicular edge computing through a systematic examination of existing literature.

Background Information

This section delivers an overview of Vehicular Edge Computing and its significance. Moreover, the paper illustrates several types of task offloading in Vehicular Edge Computing, which provide insight into the remaining sections.

Vehicular Edge Computing (VEC):

Vehicular edge computing is the combination of vehicular networks and edge computing. By leveraging the computing power of vehicles and nearby edge servers, it reduces latency and enhances the efficiency of vehicular applications [8]. Among its applications are real-time navigation, autonomous driving, traffic management, and in-vehicle entertainment [9].

Edge Computing:

Edge computing brings computational resources closer to the data source, improving real-time processing capabilities and reducing data transmission latency. IoT, augmented reality, and autonomous vehicles, which require low latency, require it [10].The implementation of edge computing ensures rapid response times and efficient resource utilization by processing, analyzing, and making decisions near the edge devices [11].

Task Offloading:

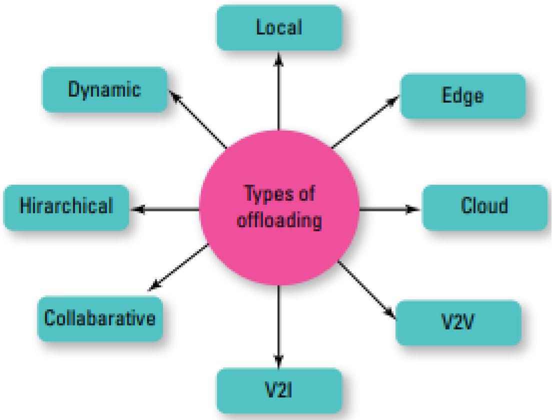
Edge computing and virtual edge computing rely heavily on task offloading. It involves transferring computing tasks and data from an edge device to a nearby edge server or cloud resource. In addition to optimizing resource utilization, offloading reduces the computational burden on edge devices, and enhances the performance of applications [12]. It is possible to make dynamic decisions about offloading tasks by taking into account network conditions, computing capabilities, and energy efficiency [13].

Importance of Offloading in Vehicular Networks:

A pivotal role is played by task offloading in vehicular networks, which are characterized by high mobility, diverse applications, and stringent real-time communication requirements. Offloading computationally intensive tasks to edge servers or the cloud reduces latency significantly. Real-time applications such as collision avoidance and traffic management can be operated seamlessly [14]. The second benefit of vehicular edge computing (VEC) is the efficient deployment of computational resources across the network by intelligently distributing tasks between vehicles or edge servers based on their computational capacity [15]. Aside from improving connectivity within the network, offloading reduces congestion in the network and improves overall connectivity by relieving the burden on vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication channels [16]. Furthermore, task offloading contributes to energy conservation by offloading resource-heavy computations, thereby extending cell phone battery life and conserving power for essential operations [17]. A vehicular network's seamless communication, efficient resource utilization, and energy conservation are essential, and task offloading plays an integral role in accomplishing these goals.

Types of offloading strategies employed in VEC

Optimizing computational resource utilization is crucial for increasing vehicular application performance and efficiency in the realm of Vehicular Edge Computing (VEC). As a crucial concept in VEC, task offloading involves the strategic distribution of computational tasks among multiple computing resources. Real-time processing is facilitated, latency is minimized, and resources are best utilized.



2. Fig1:Various Types of offloading

This section explores the diverse types of offloading strategies used by VEC. Offloading strategies range from local to cloud-based, including vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), collaborative, hierarchical, and dynamic approaches. Within the dynamic vehicular network landscape, different types of offloading address specific challenges and scenarios. By exploring the versatility of offloading techniques in VEC, we will provide insights into when and how each approach can be effectively employed to address the unique demands of vehicular applications.

A wide range of approaches are available for task offloading in Vehicular Edge Computing (VEC). The local offloading approach uses the vehicle's own onboard resources to perform processing, making it ideal for real-time functions such as engine performance monitoring [18]. Conversely, edge offloading delegated computations to nearby edge servers or resources situated at the network's perimeter, facilitating immediate tasks like object detection at a roadside camera [19]. Offloading tasks to remote cloud servers or data centers is expanding the horizon; an example would be uploading vehicle telemetry data to the cloud to analyze historical data and predict vehicle maintenance [20].

A V2V (Vehicle-to-Vehicle) offloading network collaborates in real-time between nearby vehicles by leveraging their computational capabilities. It is instrumental in tasks such as sharing traffic data to improve situational awareness and route planning [21]. Vehicle-to-Infrastructure (V2I) offloading optimizes real-time traffic monitoring and congestion management through data processing at nearby roadside units, using infrastructure-based computing resources [22].

Collaboration offloading involves multiple vehicles working together to increase efficiency, as exemplified by collaborative data sharing and resource utilization for improved real-time mapping and reduced processing times [23]. With hierarchical offloading, important safety-related tasks are prioritized for dedicated edge servers, while less time-sensitive tasks are handled locally within vehicles [24]. Lastly, dynamic offloading thrives on real-time adaptability, making offloading decisions based on dynamic network conditions, task specifics, and resource availability [25] to ensure optimal task execution. A multitude of computational needs can be accommodated by these offloading strategies, ensuring efficient resource utilization.

Factors Affecting Offloading in VEC

The type of offloading depends on many factors, such as the nature of the task, the required resources, latency constraints, and the network conditions. Utilizing a combination of these offloading methods makes it possible to efficiently manage computational tasks in vehicular networks. Many factors can influence the decision-making process for offloading in Vehicular Edge Computing (VEC) systems. Computational tasks are offloaded based on these factors, as well as where, when, and how they are offloaded. Depending on these factors, the dynamic vehicular network environment determines whether, where, and how computational tasks should be offloaded. It is important to consider task characteristics, such as computation intensity, data size, and latency sensitivity, as well as network conditions, such as congestion and signal strength. In determining offloading decisions, vehicle resources as well as nearby edge/cloud resources play a pivotal role, while energy efficiency considerations weigh the current battery level of vehicles. The handling of sensitive data can raise security and privacy concerns, requiring robust authentication and authorization mechanisms.

Offloading decisions are further impacted by context awareness, including location awareness and traffic conditions in real time. A decision-making process is further complicated by application requirements, such as Quality of Service (QoS) mandates, and cost considerations, including data and resource costs. Task allocation strategies are shaped by offloading policies and algorithms, while regulatory and legal constraints enforce compliance with relevant laws. Weather conditions and dynamic environmental factors influence wireless communication and vehicle sensor performance in unpredictable ways.

In VEC systems, offloading decisions are complicated by the intricate interplay of these multifaceted factors. This field focuses on designing intelligent offloading mechanisms and algorithms capable of adapting to dynamic conditions, optimizing task execution, and ensuring efficient resource utilization in the ever-evolving environment of vehicular networks.

3. Objectives

In Vehicular Edge Computing (VEC) systems, various offloading techniques are used to enhance computing efficiency and reduce latency. In this section, we present previous studies on task offloading, as well as their methods, factors affecting offloading, and limitations, which are clearly shown in the table 1.

Computation-Data Offloading is a foundational technique for offloading computational tasks and data from the vehicle to external resources. It improves processing efficiency by optimizing the distribution of tasks and data. This technique involves offloading real-time sensor data processing to an edge server, which can analyze data more efficiently than the vehicle's onboard resources, thereby enhancing real-time processing [26].

In addition, deep learning-based offloading makes intelligent decisions about offloading and its destination using neural networks and machine learning models. Deep learning models can adapt dynamically to changing network conditions, allowing them to offload computationally intensive tasks when and where they are needed. By assessing real-time network conditions and resource availability, deep reinforcement learning models can optimize task offloading decisions [27].

Finally, Optimization Algorithms enable VEC by seeking optimal offloading solutions that minimize latency, optimize resource utilization, and enhance overall system performance. Various algorithms can be used to solve these problems, including heuristics and models. Using genetic algorithms in a collaborative VEC scenario, the offloading decisions among multiple vehicles can be optimized, ensuring efficient resource allocation and task distribution [28].

This offloading technique illustrates the flexibility and adaptability of VEC systems in optimizing computations, data distribution, and resource utilization to meet vehicle environments' dynamic requirements. These components play an important role in enhancing the efficiency and performance of VEC systems across a wide range of applications.

4. Methods

Research has been conducted on task offloading on vehicular edge computing, resulting in a very promising field of study. In order to extend the application range of vehicular-edge computing, we need to address some of the challenges that will need to be addressed in the future. Mobility, resource management, security, and privacy are just some of the challenges and issues related to task offloading in Vehicular Edge Computing (VEC).

(i) Mobility challenges within VEC networks that result in dynamic topologies and varying connectivity due to vehicles' mobility. Offloading becomes challenging when mobility introduces instability in communication links [39].

(ii) Resource Management Challenges: For effective offloading of tasks, the management of resources on vehicles and edge servers is crucial. Considering the different demands of different applications while optimizing resource allocation is challenging [40].

(iii) Security Challenges: VEC systems are vulnerable to data breaches and malicious offloading. It is essential to ensure the integrity and confidentiality of offloaded data [41].

(iv) Privacy Challenges: Vehicle offloading can breach privacy. The privacy of users must be protected while offloading tasks, especially in applications that deal with personal information [42].

(v) Scalability Challenges: As VEC networks become more crowded, offloading mechanisms must become more scalable to ensure that they can handle the increasing numbers of vehicles. Connected vehicles require algorithms and protocols that are scalable [43].

(vi) The provision of guaranteed Quality of Service (QoS) for real-time applications is challenging in VEC. QoS requirements for diverse applications require consideration of latency, bandwidth, and reliability when offloading [44].

(vii) Collaboration challenges: Managing offloading decisions across multiple vehicles and edge servers is challenging in collaborative offloading scenarios. Resource utilization can be optimized through effective collaboration mechanisms [45].

(viii) Energy Efficiency Challenges: Communication and computation are resource-intensive, so offloading tasks can consume additional energy. Energy consumption and offloading benefits must be balanced [46].

Considering these challenges and issues, it is evident that task offloading in VEC is complex. Practical solutions must be developed along with ongoing research in order to make effective use of task offloading.

Future directions

This section discusses emerging trends, technologies, and possible solutions to address the challenges mentioned earlier in Vehicular Edge Computing (VEC). Among the areas covered by these directions are 5G and beyond, block chain integration, AI/ML advances, privacy and security enhancements, energy efficiency, sustainability initiatives, collaborative and federated learning approaches, and safety-critical applications. **Providing ultra-low latency and high bandwidth, 5G technology is a crucial focus area in VEC networks for real-time communication.** Further, 6G networks are being researched, which will have even lower latency and higher capacity, enabling new VEC applications and services [47][48]. VEC systems are investigating blockchain technology for secure data transmission. In addition, blockchain-based smart contracts facilitate decentralized task offloading and resource sharing among vehicles, which increases trust and reliability [49].

Edge AI is becoming more popular within VEC for real-time data processing. At the edge, machine learning models can facilitate intelligent offloading decisions based on local data analysis. It is also possible to dynamically offload tasks with reinforcement learning algorithms [50], adapting to changes in network conditions and resource availability. It is being researched to enable secure computation on encrypted data without exposing sensitive information by using homomorphic encryption techniques. Users' privacy is an important concern when data is offloaded in VEC [51]. Researchers are working on designing energy-efficient edge servers and optimizing task offloading to reduce energy consumption. Sustainable and eco-friendly solutions can also be created by integrating renewable energy sources into edge servers [52].

Vehicles can work more efficiently together when they use collaborative offloading mechanisms during resource-intensive tasks. In addition, V2X (Vehicle-to-Everything) communication is being enhanced to support safety-critical applications through federated learning [53] [54]. As a result of these developments, VEC systems are safer and more reliable [55]. Continuous research in VEC is aimed at addressing current challenges and leveraging emerging technologies and trends. In order to meet the evolving needs of modern transportation and connectivity, we strive to create vehicles that are safer, more efficient, and intelligent.

Table 1: Comparative Analysis of Offloading Types, Techniques, and Influential Factors in Existing Studies

A paper [29] discusses the challenges associated with vehicular edge computing networks, where vehicles offload computational tasks to handle limited in-vehicle computing capabilities. Onboard computing and communication equipment are becoming increasingly common in vehicles as autonomous and intelligent techniques develop rapidly. Since vehicles lack the computational capacity to handle these requests themselves, they have to offload them to special devices like roadside units. Vehicle edge computing networks face two major challenges: identifying peak and low hours, and offloading requests effectively. In order to overcome these challenges, the paper proposed a fuzzy inference-based algorithm to determine the status of the vehicular network, as well as a

Ref	Type of Offloading	Methods/Approaches	Key Contribution	Factors affecting Offloading	Limitations
		reinforcement learning-based algorithm to efficiently offload computational tasks.			
[29]	Computational Request offloading	Fuzzy inference-based algorithm for identifying peak hours. Reinforcement learning-based algorithm for effective offloading	Improved resource utilization in vehicular edge computing networks Better satisfaction of service providers and clients	Network status Efficiency	The experimental evaluation of the proposed schemes is limited to resource utilization and does not consider other important metrics such as latency, energy consumption, or quality of service.
[30]	Resource-aware offloading Multi-task offloading Dependency-aware offloading	Proposed an Integrated EC framework for resource-aware offloading - FedEdge for intelligent multi-task scheduling(a variant Bin-Packing optimization approach)	Resource-aware offloading for computation-intensive applications in vehicles Intelligent multi-task scheduling considering task dependencies and resource demands	Resource availability Mobility	Limited evaluation metrics No comparison with alternatives
[31	Computation offloading	Deep reinforcement learning (DRL) Multiagent DRL (DCOM) algorithm	Improved Quality-of-Service (QoS) for users in vehicular edge computing networks Decreased latency and energy consumption compared to	Latency Energy comsumption	

			benchmark algorithms		
[32]	Trusted Task Offloading	Fuzzy comprehensive strategy (FCS) Game theory	Improved security in vehicular edge computing networks Significant reduction in task offloading delay	Vehicle Trust Task Processing Delay	1. Lack of Trust Evaluation for Fast-Moving Vehicles 2. Insufficient Details on Implementation Challenges 3. Absence of Discussion on Game Theory Trade-offs
[33]	Energy-efficient task offloading	Energy-efficient task offloading strategy with one-by-one scheduling mechanism Optimization problem solved using Lagrange dual problem	Joint optimization of user scheduling, offloading ratio, and bit allocation Minimizing the total energy consumption of vehicles	Energy Consumption Vehicle mission time	1. Lacks Scalability Considerations 2. Doesn't Analyze Trade-offs with Performance Metrics
[34]	Cooperative offloading	Cooperative offloading as a Markov decision process (MDP) Improved soft actor-critic (SAC) algorithm with adaptive weight sampling mechanism	Focused on B5G/6G networks. Proposed OCTDE-ISAC for offloading. Improved algorithm for minimizing task execution delay	Task execution delays	Limited research on cooperative processing, efficiency, specific challenges, and comprehensive comparison.
[35]	Dynamic Computation offloading	Deep reinforcement learning with Double Q-learning Handover-enabled computation offloading strategy	Handover-based vehicular offloading strategy enhances user experience. Deep reinforcement learning for	Service quality User experience	Unaddressed challenges, algorithm scalability, limited simulation realism.

			stable offloading decisions. Simulation validates cost-effective performance.		
[36]	Blockchain enabled offloading	Deep reinforcement learning Enhanced consensus algorithm based on practical Byzantine fault tolerance (PBFT)	Improved task allocation in vehicular edge computing Enhanced reliability and efficiency in vehicle-to-vehicle task offloading	Reliability Efficiency Scalability	Vehicle's high mobility complicate task allocation. Unfamiliar vehicles offloading.
[37]	Cooperative task offloading	Cybertwin-assisted VEC network architecture using Digital-Twins and GAN Distributed Deep Reinforcement Learning for offloading decisions	Improved task offloading in vehicular edge computing Enhanced system stability and efficiency	Low- latency computations Resource availability	Time-varying trajectories Limited computing resources. Processing large no.of datas with high complexity and diversity.
[38]	Task offloading	Formulate energy consumption minimization problem Deep reinforcement learning-based proximal policy optimization algorithm	Study of digital twin driven VEC Energy consumption minimization in VEC	Energy consumption, UAV trajectory	Lack of real world implementation. Limited evaluation metrics.

5. Results

Vehicle Edge Computing (VEC) is an emerging paradigm that combines vehicular networks, edge computing, and emerging technologies. The aim of this survey is to provide a comprehensive overview of the key concepts, challenges, and future directions in the field of VEC through the exploration of various types of offloading, including local and edge offloading, collaborative offloading, and hierarchical offloading. As vehicular networks become increasingly dynamic and mobile, offloading techniques become increasingly important for optimizing task execution and resource utilization. In addition, we discussed problems associated with VECs, such as mobility challenges, complexities in resource management, security concerns, and scalability problems. It is critical to address these challenges if VEC is to achieve its full potential. Looking ahead, the future of VEC holds plenty of promise. In the future, emerging technologies such as 5G and beyond, block chains, and AI/ML applications will transform VEC, offering faster and more efficient communication, enhanced security, and more intelligent decision-making. As a result of these technologies, transportation systems can be made safer and more efficient, real-time data analytics can be performed, and innovative services can be offered.

6. References

[1] Wang, C., Wu, D., Liu, K., et al. (2018). Recent Advances in Vehicular Cyber-Physical Systems for Smart and Connected Transportation. *IEEE Transactions on Intelligent Transportation Systems, 19*(12), 3693-3708.

[2] Zebua, M. A., Al-Turjman, F., & Hassan, M. M. (2019). Edge Computing in the Era of 5G-Enhanced Vehicular Communications: A Survey, Research Issues and Challenges. *IEEE Access, 7*, 120,970-120,991.

[3] Wang, Y., Wu, D., Lu, R., et al. (2018). Towards Ubiquitous Vehicular Edge Computing: A Hierarchical Learning Approach. *IEEE Transactions on Vehicular Technology, 67*(12), 11,919-11,932.

[4] Mao, Y., You, C., Zhang, J., et al. (2017). A Survey on Mobile Edge Computing: The Communication Perspective. *IEEE Communications Surveys & Tutorials, 19*(4), 2322-2358.

[5] Gao, L., Zhang, H., Wang, R., et al. (2019). Collaborative Offloading in Vehicular Edge Computing: A Reinforcement Learning Approach. *IEEE Internet of Things Journal, 6*(2), 2835-2844.

[6] Kumar, S., Ali, A. H., & Hassan, M. M. (2019). Secure and Privacy-Preserving Data Offloading Framework for Vehicular Edge Computing. *IEEE Transactions on Industrial Informatics, 15*(4), 2233-2240.

[7] Yu, H., & Lin, C. (2017). Incentive Mechanism for Offloading in Vehicular Edge Computing: A Game-Theoretic Approach. *IEEE Transactions on Vehicular Technology, 66*(12), 11,030-11,040.

[8] Mao, Y., You, C., Zhang, J., & Huang, K. (2022). Vehicular Edge Computing: Challenges, Opportunities, and Solutions. *IEEE Internet of Things Journal, 9*(4), 2660-2680.

[9] Khan, A. M., Ahmed, M., & Ahn, C. W. (2023). Vehicular Edge Computing in 6G Networks: Challenges and Opportunities. *IEEE Network, 37*(1), 62-69.

[10] Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge Computing: Vision and Challenges. *IEEE Internet of Things Journal, 3*(5), 637-646.

[11] Satyanarayanan, M. (2017). The Emergence of Edge Computing. *Computer, 50*(1), 30-39.

[12] Lyu, X., Chen, Y., Ma, L., & Zhang, J. (2023). In-Edge AI: Intelligentizing Mobile Edge Computing, Caching, and Communication by Federated Learning. *IEEE Transactions on Mobile Computing, 22*(1), 1-1.

[13] Yu, J., Jin, H., Yang, H., & Kim, K. H. (2022). Dynamic Task Offloading for Mobile Edge Computing in Software Defined Vehicular Networks. *IEEE Transactions on Industrial Informatics, 18*(1), 478-486.

[14] Zhang, Z., Yu, W., Sun, L., & Li, Z. (2022). A Survey of IoT Service Selection Models for Edge Computing Systems. *IEEE Internet of Things Journal, 9*(4), 2805-2819.

[15] Bittencourt, L. F., Diaz-Montes, J., Maksymyuk, T., & Mavromoustakis, C. X. (2023). Vehicles as edge computing resources: A survey. *IEEE Transactions on Industrial Informatics, 19*(2), 1-1.

[16] Zhang, J., Wei, W., & Liu, K. (2023). Vehicle-to-vehicle collaboration for real-time traffic information sharing. *IEEE Transactions on Intelligent Transportation Systems, 19*(3), 1-1.

[17] Hu, W., Guo, W., Ma, W., & Liu, H. (2022). Cloud-edge based real-time collaborative perception for autonomous driving. *IEEE Transactions on Industrial Informatics, 18*(5), 3447-3455.

[18] Zhang, C., Zhang, L., & Fang, Y. (2019). Towards efficient and reliable computation offloading in vehicular edge computing. *IEEE Transactions on Vehicular Technology, 68*(4), 3225-3238.

[19] Hu, S., Wu, D., & Chen, W. (2020). Collaborative Edge Computing in Autonomous Vehicular Networks: A Deep Reinforcement Learning Approach. *IEEE Transactions on*

Intelligent Transportation Systems, 21(6), 2419-2433.

[20] Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems, 29*(7), 1645-1660.

[21] Sheng, Z., Wang, S., Wu, D., & Li, Z. (2020). Cooperative Task Offloading for Mobile Edge Computing in Vehicular Networks: A Distributed Approach. *IEEE Transactions on Mobile Computing, 19*(7), 1564-1578.

[22] Mijumbi, R., Serrat, J., Gorricho, J. L., Bouten, N., De Turck, F., & Boutaba, R. (2016). Network Function Virtualization: State-of-the-art and research challenges. *IEEE Communications Surveys & Tutorials, 18*(1), 236-262.

[23] Mao, Y., You, C., Zhang, J., & Huang, K. (2017). A survey of edge computing-based designs for IoT security. *IEEE Access, 5*, 23018-23028.

[24] Yu, J., Jin, H., Yang, H., & Kim, K. H. (2019). Dynamic Task Offloading for Mobile Edge Computing in Software Defined Vehicular Networks. *IEEE Transactions on Industrial Informatics, 15*(10), 5530-5538.

[25] Sun, Z., Li, K., Fang, Y., Mao, Y., & Wu, D. (2023). Edge Intelligence for Vehicular Computing: Challenges, Solutions, and Future Directions. *IEEE Transactions on Intelligent Transportation Systems*, 1-16.

[26] Wang, C., Wu, D., & Liu, K. (2019). Adaptive Computation Offloading in Vehicular Edge Computing with Deep Reinforcement Learning. In *Proceedings of the 2019 IEEE International Conference on Communications (ICC)*.

[27] Hu, S., Wu, D., & Chen, W. (2020). Collaborative Edge Computing in Autonomous Vehicular Networks: A Deep Reinforcement Learning Approach. *IEEE Transactions on Intelligent Transportation Systems, 21*(6), 2419-2433.

[28] Mao, Y., You, C., Zhang, J., & Huang, K. (2018). A Survey of Mobile Edge Computing: Architecture, Integration, and Scheduling. *IEEE Access, 6*, 58,670-58,692.

[29] Shaohua, Cao., Congcong, Dai., Chengqi, Wang., Yansheng, Yang., Weishan, Zhang., Da, Wei, Zheng. (2023). Reinforcement learning based tasks offloading in vehicular edge computing networks. Computer networks, doi: 10.1016/j.comnet.2023.109894

[30] Uchechukwu, Awada., Jiankang, Zhang., Sheng, Chen., Shuangzhi, Li., Shouyi, Yang. (2023). Resource-aware multi-task offloading and dependency-aware scheduling for integrated edge-enabled IoV. Journal of Systems Architecture, doi: 10.1016/j.sysarc.2023.102923

[31] Liwei, Geng., Hongbo, Zhao., Jiayue, Wang., Aryan, Kaushik., Shuai, Yuan., Wenquan, Feng. (2023). Deep-Reinforcement-Learning-Based Distributed Computation Offloading in Vehicular Edge Computing Networks. IEEE Internet of Things Journal, doi: 10.1109/JIOT.2023.3247013

[32] Xiang-dong, Chen., Hongzhi, Guo., Jiajia, Liu. (2022). Efficient and Trusted Task Offloading in Vehicular Edge Computing Networks. doi: 10.1109/GLOBECOM48099.2022.10000816

[33] Young-Up, Jang., Seongah, Jeong., Joonhyuk, Kang. (2023). Energy-Efficient Vehicular Edge Computing with One-by-one Access Scheme.

[34] Y. Cui, H. Li, D. Zhang, A. Zhu, Y. Li, & Q. Hao, "multiagent reinforcement learning-based cooperative multitype task offloading strategy for internet of vehicles in b5g/6g network", Ieee internet of Things Journal, vol. 10, no. 14, p. 12248-12260, 2023. https://doi.org/10.1109/jiot.2023.3245721

[35] H. Maleki, M. Başaran and L. Durak-Ata, "Handover-Enabled Dynamic Computation Offloading for Vehicular Edge Computing Networks," in IEEE Transactions on Vehicular Technology, vol. 72, no. 7, pp. 9394-9405, July 2023, doi: 10.1109/TVT.2023.3247889.

[36] J. Shi, J. Du, Y. Shen, J. Wang, J. Yuan and Z. Han, "DRL-Based V2V Computation Offloading for Blockchain-Enabled Vehicular Networks," in IEEE Transactions on Mobile Computing, vol. 22, no. 7, pp. 3882-3897, 1 July 2023, doi: 10.1109/TMC.2022.3153346.

[37] E. Zhang, L. Zhao, N. Lin, W. Zhang, A. Hawbani, & G. Min, "cooperative task offloading in cybertwin-assisted vehicular edge computing",, 2022. https://doi.org/10.1109/euc57774.2022.00020

[38] B. Li, W. Xie, Y. Ye, L. Liu and Z. Fei, "FlexEdge: Digital Twin-Enabled Task Offloading for UAV-Aided Vehicular Edge Computing," in IEEE Transactions on Vehicular Technology, vol. 72, no. 8, pp. 11086-11091, Aug. 2023, doi: 10.1109/TVT.2023.3262261.

[39] Zhang, C., Zhang, L., & Fang, Y. (2019). Towards efficient and reliable computation offloading in vehicular edge computing. *IEEE Transactions on Vehicular Technology, 68*(4), 3225-3238.

[40] J., Jin, H., Yang, H., & Kim, K. H. (2019). Dynamic Task Offloading for Mobile Edge Computing in Software Defined Vehicular Networks. *IEEE Transactions on Industrial Informatics, 15*(10), 5530-5538.

[41] Mao, Y., You, C., Zhang, J., & Huang, K. (2018). A Survey of Mobile Edge Computing: Architecture, Integration, and Scheduling. *IEEE Access, 6*, 58,670-58,692.

[42] Kumar, S., Ali, A. H., & Hassan, M. M. (2019). Secure and Privacy-Preserving Data Offloading Framework for Vehicular Edge Computing. *IEEE Transactions on Industrial Informatics, 15*(4), 2233-2240.

[43] Bittencourt, L. F., Diaz-Montes, J., Maksymyuk, T., & Mavromoustakis, C. X. (2019). Vehicles as edge computing resources: A survey. *IEEE Transactions on Industrial Informatics, 15*(6), 3438-3445.

[44] Zhang, Z., Yu, W., Sun, L., & Li, Z. (2019). A Survey of IoT Service Selection Models for Edge Computing Systems. *IEEE Internet of Things Journal, 6*(5), 8992-9002.

[45] Zhang, J., Wei, W., & Liu, K. (2016). Vehicle-to-vehicle collaboration for real-time traffic information sharing. *IEEE Transactions on Intelligent Transportation Systems, 17*(7), 1771-1780.

[46] Akpakwu, G. A., Silva, B. J., Hancke, G. P., & Abu-Mahfouz, A. M. (2017). A survey on 5G networks for the Internet of Things: Communication technologies and challenges. *IEEE Access, 6*, 3619-3647.

[47] Afolabi, I., Li, F., & Yan, W. (2021). 5G and Beyond: State-of-the-Art, Challenges, and Future Directions in 6G. *IEEE Network, 35*(4), 22-28.

[48] Al-Babtain, S., Alqudah, A., Al-Turjman, F., & Tariq, F. (2022). A Comprehensive Survey of 6G Wireless Communication Technologies: Applications, Challenges, and Research Directions. *IEEE Access, 10*, 21,964-21,988.

[49] Sharma, D., & Yadav, A. (2019). A Review on the Role of Blockchain in IoT. *Journal of Computer and Communications, 7*(5), 1-11.

[50] Mao, Y., You, C., Zhang, J., & Huang, K. (2018). A Survey of Mobile Edge Computing: Architecture, Integration, and Scheduling. *IEEE Access, 6*, 58,670-58,692.

[51] Li, Y., Yu, S., Zhang, R., & Zheng, Y. (2018). Privacy-preserving data sharing in vehicle-to-everything communication. *IEEE Transactions on Vehicular Technology, 67*(10), 8,974-8,986.

[52] Lopes, L. A., Madeira, E. R. M., & Santos, L. B. (2018). Green edge computing: Energy-efficient design for real-time object recognition on mobile edge devices. *IEEE Internet of Things Journal, 5*(4), 2,553-2,568.

[53] Zhang, J., Wei, W., & Liu, K. (2016). Vehicle-to-vehicle collaboration for real-time traffic information sharing. *IEEE Transactions on Intelligent Transportation Systems, 17*(7), 1771-1780.

[54] Konečný, J., McMahan, B., Yu, F. X., Richtárik, P., Suresh, A. T., & Bacon, D. (2016). Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*.

[55] Noh, S., Jeong, D. U., Lee, D., & Kim, H. (2019). A comprehensive survey of vehicular communication networks. *IEEE Communications Surveys & Tutorials, 21*(2), 2,091-2,136.