A review of computational offloading research based on game theory in a vehicular edge computing environment

Zijian Qu[a], Yinghua Tong[a,b,*], Maocuo Li[a]
[a]School of Computer, Qinghai Normal University, Xining 810008, China; [b]The State Key Laboratory of Tibetan Intelligent Information Processing and Application, Qinghai Normal University, Xining 810008, China

ABSTRACT

A new computing paradigm that extends the capabilities of the cloud to the edge of the network, vehicular edge computing (VEC), provides vehicles with a wealth of resources that can accommodate the rising demand for in-vehicle application computing and storage resources. This is accomplished by seamlessly fusing mobile edge computing with in-vehicle networks (MEC). It provides low-latency, high-bandwidth, and extremely dependable computing services to vehicles and other users with limited computing resources. This paper reviews existing game theory-based task offloading mechanisms in the vehicle edge environment. The first section introduces the current research status of vehicle edge computing in terms of its features, architecture, and applications. It covers the background and significance of task offloading in vehicle edge computing, the VEC architecture, the advantages of VEC, and several alluring application scenarios. A thorough analysis of the body of prior research is then presented, followed by a categorical discussion of game theory-based computational offloading mechanisms in the context of vehicle edges. Finally, predictions for the future of technology are made.

Keywords: Mobile edge computing, vehicular edge computing, task offloading, game theory, resource allocation

1. INTRODUCTION

As Intelligent Transportation Systems (ITS)[1] and the Internet of Vehicles (IoV) advance and more vehicles become intelligent, in-vehicle networks are piquing both academic and commercial interest. In-vehicle video streaming, image-assisted navigation, intelligent vehicle control, and online gaming are just a few of the new, complex in-vehicle applications made possible by advancements in hardware and software that have increased the processing and communication capabilities of vehicles[2]. These applications have stringent latency requirements while handling complex data processing and storage operations that require significant computing and storage resources[3-4].

The first method used by the global community to address computing and storage issues is cloud computing[5]. The concept of customer-focused cloud computing was developed in anticipation of the transition of computer infrastructure to paid computing and storage services offered via the Internet [6]. Applications that cannot execute on local hardware can be uploaded to the cloud over the Internet. In-vehicle networks and cloud computing are coupled to satisfy task execution requirements and make greater use of in-vehicle network resources. These applications are most commonly associated with traffic control and video surveillance [7].

One solution is to transfer a vehicle's computing needs to a distant cloud server. Although using cloud storage and computing on devices with limited resources has many advantages, there are several important factors to take into account when offloading computing to the cloud. The primary disadvantages of the cloud computing paradigm are the distance from remote cloud servers, the significant dependency on the Internet, and the backhaul network congestion [8]. Efficiency is impacted by an increase in the time required to offload data to the cloud due to increasing user data generation rates and the distance between cloud servers and users. As a result, data processing can be done more effectively at the network's edge. Edge computing is a method of computing that relocates data processing to the edge of the network [9-10]. The term "mobile edge computing" (MEC) describes the use of these edge computing servers for nearby mobile phones and vehicles, as well as other mobile computing devices [11-12]. A typical mobile edge computing framework is represented in Figure 1.

*tyh_angel@126.com

Seventh International Conference on Mechatronics and Intelligent Robotics (ICMIR 2023), edited by Srikanta Patnaik, Tao Shen, Proc. of SPIE Vol. 12779, 127791I · © 2023 SPIE · 0277-786X · Published under a Creative Commons Attribution CC-BY 3.0 License · doi: 10.1117/12.2688679

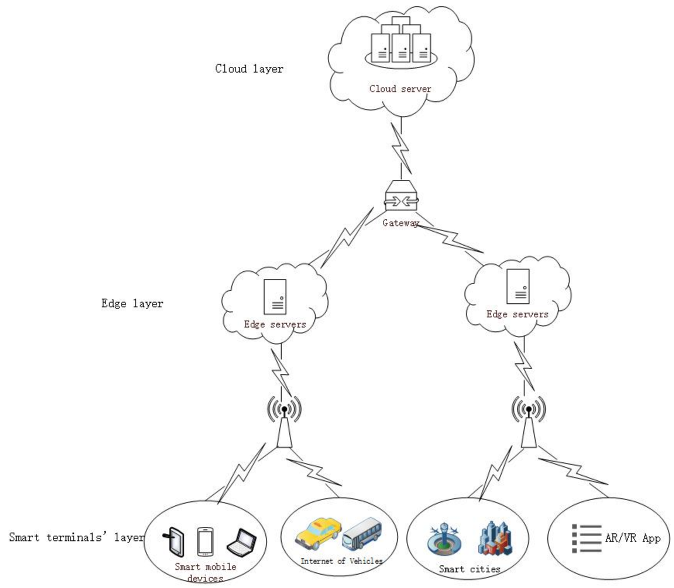

Figure 1. A typical mobile edge computing framework.

Traditional cloud computing is no longer advantageous in the context of telematics for handling data-intensive and latency-sensitive computing tasks in vehicles. Therefore, in order to move the processing of computational tasks to the edge of the vehicle network, a new computing paradigm known as Vehicular Edge Computing (VEC) has emerged in this scenario. Compute offloading, which enables resource-constrained vehicles to transfer tasks to other nodes with enough resources to expedite processing and raise service quality, is a significant application of VEC. In VEC, computational tasks generated by vehicle applications can be offloaded to fixed infrastructures such as Road Side Units (RSUs). To accommodate the computational requirements of vehicles within their communication range [13-14], on the other hand, underutilized storage and computational resources in vehicles can be thought of as "edge" infrastructure [15-16]. Offloading decisions and computational resource allocation are the two main issues in computational offloading. Offloading decisions consider whether and how much the mobile terminal needs to be offloaded. The focus of computational resource allocation, in contrast, is on where to offload the necessary computational tasks [17]. The decision to offload concerns whether and how much should be done so, whereas the decision to allocate computing resources concerns where the necessary computing tasks should be offloaded.

Recent years have seen a significant increase in research interest in task offloading by MEC and VEC. In order to solve offloads from mobile devices to edge servers, Wang et al.[18] categorized existing offload solutions using five dimensions: offload target, EC load balancing, user device mobility, application partitioning, and partition granularity, and then thoroughly analyzed their mathematical models. In their study [19], Mach et al. examine the user-oriented use case of compute offloading from three perspectives: compute offloading choices, compute resource allocation inside the MEC, and mobility management. In addition, they provide the primary use cases and reference situations for MEC. Gu et al.[20] outlined the basic concepts and architecture of MEC and typical application scenarios, then introduced the application of MEC in in-vehicle networks, the current status of research on MEC in in-vehicle networks based on software-defined networks (SDN), and examples of MEC applications in in-vehicle networks, and gave the problems and challenges to be faced in deploying MEC in in-vehicle networks. A survey on vehicular networks and computational offloading is presented by Boukerche et al. in[3]. The analysis considers the case of a car acting as an edge node for a mobile device with constrained computing capabilities, concentrating on computational offloading solutions and categorizing them in accordance with partitioning, scheduling, and data retrieval methods. However, no research has been conducted to summarize and analyze the problems associated with task offloading in VEC environments employing game theory (GT) mechanisms. This paper provides a comprehensive summary of a number of survey papers concerning GT-based offloading strategies in VEC. To do this, we look at the most recent GT-based methods used in VEC offloading based on research that has already been done.

In Section 1, the paper introduces the background and significance of computational task offloading in VEC environments and the current status of related research. Section 2 introduces the system architecture of VEC, the advantages of VEC, and several attractive application scenarios. Section 3 provides a detailed introduction and categorization of existing game theory research in VEC environments. Section 4 summarizes several challenges associated with VEC task offloading and suggests future directions for research. The final section concludes the paper.

2. THE ARCHITECTURE OF VEHICLE EDGE COMPUTING AND ITS BENEFITS

2.1 Architecture of vehicle edge computing

Figure 2 depicts the VEC's architecture. The architecture typically consists of three layers: the user layer (the vehicle terminal), the MEC layer (the RSU), and the cloud layer (the cloud server).

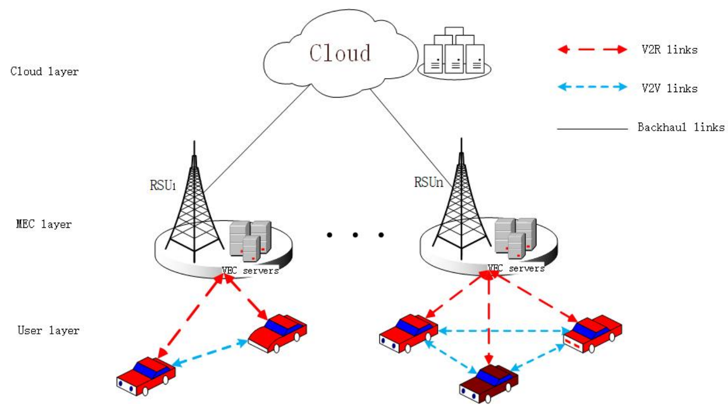

Figure 2. The architecture of vehicle edge computing.

1)Vehicle-Mounted Terminal: The primary representation of the vehicle-mounted terminal in VEC is a vehicle. The following standout characteristics of vehicles are when compared to standard mobile nodes:

(i)Sensing: Utilizing onboard equipment with cameras, radar, GPS, and other onboard devices, vehicles can sense their surroundings both internally and externally.

(ii) Communication: Vehicles can communicate with other vehicles and RSUs using the V2V (vehicle to vehicle) and V2R (vehicle to RSU) protocols, as shown in Figure 3.

(iii) Computing: Vehicles can process some of the computing tasks locally in addition to sending some of them to edge servers or the cloud.

(iv)Storage: Popular content for data sharing can be cached on the vehicle's unused storage space.

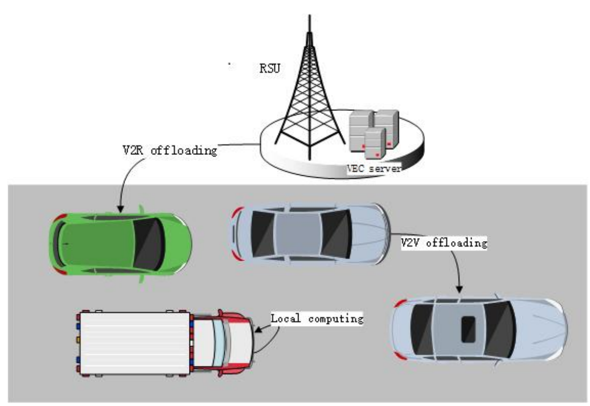

Figure 3. Example of vehicle task offloading communication in vehicle edge computing.

2)Edge servers: RSUs are typically situated alongside city streets and serve as edge servers in the VEC. Compared to vehicles, they have an abundance of communication, computing, and storage resources. Receiving data from vehicles, processing it, and even uploading it to the cloud are all tasks that fall under the purview of RSUs. RSUs make it easier to handle demanding performance requirements with computational offloading and caching technology. Additionally, they can offer the vehicle a variety of services like video streaming, traffic management, and route navigation.

3)Cloud servers: Remote clouds are used to deploy cloud services. They can use the edge servers to access the uploaded data. Cloud services have a larger coverage area and abundant computing and storage capacity when compared to edge servers. By gathering uploaded data from mobile nodes and edge servers, they are able to give a comprehensive view of the coverage area. Making the best decisions is made easier with the help of centralized management and global control offered by cloud computing.

2.2 Vehicle edge computing benefits

2.2.1 Response time and ultra-low latency advantages

Response time and ultra-low latency: Response time includes the time it takes for data to be offloaded to an edge server, returned, and processed. Edge servers in a VEC are closer to the vehicle user than cloud servers. As a result, they enable the delivery of services with extremely low latency and high reliability while requiring much less processing time. For latency-sensitive applications like autonomous driving [21] and road safety, this is especially advantageous.

In autonomous driving, vehicles must have a precise and timely awareness of their surroundings and carry out more effective and efficient operations. Thanks to VEC, autonomous driving can now be supported by robust computing resources to carry out its tasks. An edge server close to the vehicle can analyze and process this data in real-time after receiving it directly from the vehicle and sensors placed along the road. The edge server is in charge of alerting surrounding vehicles when risk data is found and taking reasonable precautions to avoid potential hazards, such as braking, lane changes, and U-turns, thereby enhancing road safety.

2.2.2 Bandwidth and storage advantages

Bandwidth and storage: The volume of data produced by smart vehicles will soar as they become more prevalent. The requests for content will also diversify. As a result of the great distances between users and the cloud, cloud computing cannot satisfy the bandwidth needs for large-scale data processing and content delivery through centralized administration. By relocating the cloud's computation and storage resources to the network's edge, VEC is able to effectively alleviate the enormous bandwidth strain on the backhaul network. Additionally, unlike the cloud, data can be kept on edge servers in the VEC that are close to the vehicle user. Vehicle users can quickly access stored data thanks to caching technology, which also eases the load on remote clouds in terms of storage.

Developments in bandwidth and storage technology have freed drivers from the complexities of driving, and in-car entertainment has become more diverse, including surfing the internet, playing games, watching videos [22], and so on. When it comes to watching videos, in-vehicle users can access requested videos directly without the need for a remote cloud, reducing latency and enhancing the user experience. This is accomplished by cooperatively caching popular content between edge servers and vehicles.

2.2.3 Proximity services and contextual information

Proximity services and contextual information: Numerous proximity services can be offered to users as a result of the edge server's proximity to the VEC's vehicle users. This improves the user experience while still maintaining efficient traffic management. For instance, the edge server can assist in processing location and sensory data from vehicles to create high-definition (HD) maps, which can then be sent to the vehicle. The edge server in the VEC has access to real-time data about the location and behavior of the vehicle, as well as data about the network and traffic conditions. Numerous applications can be improved with the help of this knowledge.

For instance, real-time navigation systems require data sensing, acquisition, and processing to be implemented. For the real-time navigation system in the VEC, the edge server can offer crucial computing and storage resources. The communication area is covered by the edge server. Each vehicle in this area transmits to the appropriate server its current status (position, speed), as well as the data it has gathered (weather, road conditions). The server receives the data that the vehicle uploads and uses that information to find out about the local traffic conditions and manage the flow of traffic.

Traffic congestion can be prevented in this way[23]. The best routes are made available to drivers thanks to this, which is very advantageous.

3. GAME THEORY-BASED COMPUTATIONAL OFFLOADING MECHANISM IN THE VEC ENVIRONMENT

An interdisciplinary model called GT is used to describe significant interactions in situations involving multiple decisions and rational participant behavior[24]. Typically, GT in computational offloading is a classical tool for modeling resource allocation problems. A traditional game with n players consists of three elements (N, A, u), where N is the set of players N = (N1, ..., Nn), A is the set of actions taken by these players, A = (A1, ..., Ai). The m available actions of the player are defined as Ai = (a1, ..., am), and u = (u1, ..., un) is the player's payoff function. In the beginning, players decide based on the action they choose. Finally, the payoffs (penalties or rewards) that players receive are influenced by the decisions of the other players.

The GT architecture is extensively used in wireless communications and networks as well as a method of computational offloading due to its built-in planning behavior[25]. The GT approach has also been widely studied and applied in computational offloading at the vehicle edge[26-28]. Entities that are already part of a network aren't likely to share their available resources with other nodes for free because of their natural self-interest. Game theory is a mathematical theory and methodology that looks at how to find the most rational solution and the most rational behavioral response for the parties who are having trouble sharing resources in a game. As a result, to encourage them to contribute resources, a suitable incentive scheme is required. Once the resources are shared, they should be rewarded, and user behavior with regard to shared network resources can be examined in order to meet their needs.

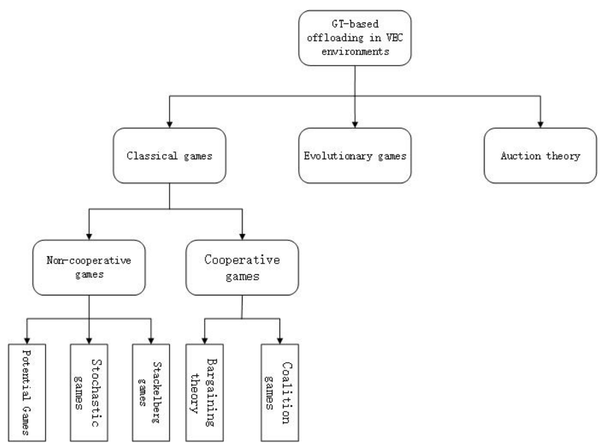

Figure 4. GT-based computational offloading algorithm in vehicle edge computing.

The three main categories of GT-based algorithms presented in the existing literature are classical, evolutionary, and auction forms, as shown in Figure 4. A traditional game is a static strategy in which players make logical decisions based on predetermined action options. However, because they have little knowledge about the consequences of their actions, players in evolutionary games are only partially rational. In an auction theory scenario, participants bargain to sell or purchase the desired service for a reasonable price because they are unsure of the value of their own services.

In addition, cooperative and non-cooperative games are separated into two subcategories[24]. The former applies better in cooperative settings where fairness is valued more highly. In this scenario, various players cooperate to achieve their shared objectives and goals to the fullest extent possible. The latter emphasizes individual players' games in which they disregard one another's objectives.

3.1 Non-cooperative based GT offloading mechanism in VEC

Typically, non-cooperative games attempt to reach either the Stackelberg or Nash equilibrium as a solution. The decision-making process of a non-cooperative game involves many players with common interests. All participants make decisions individually; each participant is self-interested, and all participants compete with each other to maximize their interests, which means that there is an optimal strategy to maximize the benefits of the system when the model has the Nash equilibrium property[29].

Non-cooperative games can be divided into a number of subcategories, some of which are not covered in the articles under study because they are a relatively well-established genre. The most prevalent game theories and the most well-liked research areas are potential, stochastic, and Stackelberg. For instance, a lot of mobile users try to send their resource-intensive tasks over a wireless channel to a remote server. In this illustration, every user has competing interests and wants to spread their content to increase their own productivity. This competitive and self-centered approach increases interference in the computing paradigm as a whole, which hurts the performance of the system as a whole and, in turn, hurts the performance of each user.

Ye et al.[16] worked on a new workload allocation paradigm that proposes dividing vehicle terminals into resource provisioning terminals (RPTS) and resource demand terminals (RDTS) in order to increase the performance of VEC servers (RDTS). In this system, a sequential Stackelberg game is used to build an efficient workload allocation method. VEC servers, RDTs, and RPTs can effectively coordinate workload allocation to achieve a Nash balance between the two sequences with the aid of the sequential Stackelberg game.

Zhang et al.[28] proposed a tiered cloud-based offloading framework for mobile edge computing (MEC), where a backup compute server is added close by to make up for the MEC server's lack of computing power. This framework is used to design the best multi-level offloading scheme, which maximizes the use of the offloading service providers as well as the use of the vehicles and computing servers.

Zeng et al.[30] investigated the most effective and cost-effective ways to use idle resources in volunteer vehicles to manage busy workloads on VEC servers. The proposed model for volunteer-assisted vehicle edge computing defines cost and utility functions for the requesting vehicle and the VEC server. Receiving a reward from the overburdened VEC server encourages the volunteer vehicle to help. The interaction between the requesting vehicle and the VEC server is then examined using the Stackelberg game. Theoretically, it is shown that the Stackelberg game between the VEC server and the requesting vehicle has a special Stackelberg equilibrium. To select the optimal pricing plan for the VEC server, it is recommended to apply a rapid search technique based on a genetic algorithm. A volunteer task allocation method is also proposed to create the best match between tasks and volunteer alliances, lower the cost of unloading vehicles, and make the VEC server more useful so that volunteer vehicles get the most reward.

Mobile Edge Computing (MEC) and the Internet of Vehicles (IoV) were combined by Li et al[31]. The FOG nodes (FNS), Data Service Agents (DSAS), and CARs make up the proposed specific resource management framework for the vehicle edge. The workload on the DSA is intended to be balanced and the service level increased by a dynamic service area allocation algorithm. It is suggested that a distributed iterative algorithm and a framework for allocating resources based on the Stackelberg game model be used to look at the pricing problem of the FNS and the data resource strategy of the DSA. The suggested framework guarantees effective FN resource distribution among cars, optimal participant strategies, and perfect Nash balance.

Zhang et al.[32] built a MEC network architecture for in-vehicle networks using software-defined networks (SDN). An SDN controller is used to sense the network state on a global scale. To further reduce system overhead, offloading and resource allocation problems for V2X are proposed. Optimal offloading choices, transmission power control, sub-channel allocation, and computational resource allocation schemes are provided. Three phases make up the offloading task optimization problem. First, hierarchical analysis is used to choose the initial offloading nodes. Then, transmission power, sub-channels, and computational resources are allocated using stateless Q-learning; additionally, the decision about offloading is modeled as a potential game, and a potential function is built to show that there is a Nash equilibrium.

Pham et al.[33] investigated a partial offloading technique for multi-user parked vehicle-assisted multi-access edge computing (PVMEC), in which tasks from each mobile device (MD) can be partially offloaded to the MEC server or an adjacent parked vehicle (PV). Consider the offloading decision, the offloading ratio, and resource allocation while formulating the system utility maximization problem. Taking into account the complexity and privacy concerns of centralized schemes and the inherent resource contention among MDs, a low-complexity distributed offloading scheme is designed using a game-theoretic approach to model the offloading decision problem as an exact potential game and a sub-gradient method to determine the optimal offloading ratio and resource allocation.

Zhang et al.[34] developed an end-edge cloud collaboration paradigm in order to utilize vehicles with idle resources as fog user devices (F-UEs) for computation in a fogged vehicle network (FVNET). A system for adaptive dual-timescale resource reservation and allocation is provided in order to coordinate end-edge cloud resources under varied demand conditions. The cloud server is first proposed as the dominant player in a dynamic F-UE incentive problem based on the Stackelberg game, with multiple F-UEs acting as followers. Then, an iterative approach is proposed to achieve Stackelberg equilibrium in the pricing and reservation of computer resources. The issues of collaborative communication and computational resource allocation are shifted to a multi-intelligent stochastic game with a shorter time horizon. To lower overall latency, a loosely distributed approach based on multi-intelligent deep reinforcement learning is created. The F-UE incentive optimization is activated, and more F-UE resources are reserved as the latency performance improves.

3.2 Cooperation-based GT offloading mechanism in VEC

Cooperation between logical players is facilitated in cooperative games by pre-established agreements. The utility of the users' choices is impacted by this cooperation. Broad and implicit rules, a focus on coalitions, and the availability of commitments are characteristics of this type of GT[35].

Typically, coalition games and bargaining theory games are the two main subcategories of cooperative games. According to bargaining theory, parties with competing interests and limited resources can come to a compromise that benefits both parties without coercion[25]. For edge-assisted mobile crowdsensing, Liu et al.[36] presented a vehicle recruitment system that considers incentives. Through the application of an incentive scheme and Nash bargaining theory, cooperation between edge servers and intelligent vehicles is fostered. In addition, a realistic and effective accounting system for vehicle contributions is proposed. Then, it is established that vehicle participant recruiting is an NP-hard optimization problem. Utilizing submodular optimization features, an efficient heuristic approach with a guaranteed approximation ratio is proposed for this problem.

Coalition games focus on the creation of user-cooperative groups. Even when these agreements are not explicitly stated or implied by the rules, coalition games are cooperative if agreements can be reached to divide the rewards among the players[24]. Regarding the issue of popular content distribution (PCD) for in-vehicle units, Chen et al.[37] proposed a cooperative V2V-assisted transmission scheme based on coalition gaming (CVCG). The plan enables the in-vehicle unit to collaborate with nearby devices to deliver popular content that is missing. A coalition graph gaming algorithm is also created to optimize OBUs' cooperative behavior. There has been significant progress made in addressing the issue of popular content being received and distributed by in-vehicle units in poor quality due to their high speed, constrained bandwidth, and erratic wireless connectivity.

3.3 Evolutionary-based GT offloading mechanism in VEC

Evolutionary games differ from traditional game theory in that the idea of evolutionary games is inspired by the evolution of living organisms. In games where the players are not completely rational, each player gradually adjusts to reach equilibrium by analyzing changes in conditions over time that may be caused by irrational behavior. Evolutionary games are mostly used in multi-user computing offload scenarios, where each device is unaware of the other players' strategies in a dynamic environment and where the players continuously adjust and improve their strategies to maximize the group's benefits [38].

The coordination issue between multiple vehicles in route planning algorithms was studied by Lu et al.[39]. Vehicle coordination is proposed using a distributed cooperative routing algorithm (DCR) based on evolutionary game theory. The process is implemented on a roadside unit (RSU) using edge computing and edge intelligence. In order to assess the effectiveness of the suggested algorithm, road networks are constructed.

3.4 Auction theory-based offloading mechanism in VEC

The economic theory of auctions, originally attributed to the sciences of economic theory and operations research, is one of the oldest price discovery mechanisms. Auction theory is also an effective method for studying the allocation and pricing of VEC resources. In the theory of auctions, the participants' strategy is typically to offer the item for sale at the best price at which neither party is certain of its value [40]. These objects can be categorized as various computing and communication services that require the user's consent under the offloading criterion.

According to a Vehicle Edge Computing (VEC) caching scheme put forth by Wang et al.[41], content providers (CPs) cooperatively cache popular content in parking storage facilities spread across various parking lots. The proposed VEC caching scheme expands the data center's capabilities from the network's core to its edge. Duplicate transfers from distant servers can be stopped as a result, and overall transfer latency can be greatly decreased. To cut down on the average latency for mobile users, it is suggested that a content placement algorithm based on an iterative rising price auction could be used.

A novel security system was developed by Hui et al.[42] for sixth-generation (6G) heterogeneous vehicular networks in order to provide mobile vehicles (MVs) with individualized edge computing services (HetVNETs). The plan develops a secure edge computing architecture based on smart contracts by taking into consideration both attack models and the features of 6G network infrastructures like satellites, drones, base stations, and roadside units, each of which manages multiple parked vehicles to collaborate on computing services. According to this architecture, a collaborative computing resource allocation algorithm is developed based on the computing resources that are currently available and owned by different network infrastructures to help each network infrastructure choose a customized service strategy (CSS) to meet the QoE of MVs. Following the determination of the CSS, a model based on a second-price sealed auction is built to depict the rivalry between network infrastructures and identify the game's Nash equilibrium, which will guide each infrastructure's best strategy for bidding for the opportunity to complete the service.

Zhang et al.[43] proposed deploying edge computing nodes to support blockchain technology and implementing an auction mechanism to motivate users to record driving data as miners. To optimize societal welfare, this study offers two auction techniques for the blockchain network, which is comprised of edge computing service providers and miners. In particular, one technique is utilized when the resource needs of the miners are the same, while the other is used when the resource demands of the miners are different. For optimum allocation of resources, the former utilizes the maximum cost-maximum flow method, whereas the latter employs a heuristic approach. The former employs the Vickrey–Clarke–Groves method for price payment, whereas the latter uses dichotomy. These two payment algorithms compute the payment price using critical value theory. This study shows that both are true and individually reasonable. This article assesses measures such as social welfare, satisfaction, and resource use using experiments. Experiments demonstrate that the suggested auction method may successfully optimize social welfare in a blockchain network and offer an efficient resource allocation approach for service providers of edge computing.

4. FUTURE RESEARCH DIRECTIONS

(1) Mobility

High-speed vehicle networks are frequently subjected to changes in the traffic-flow topology because of their dynamic nature. The neighbors of moving vehicles on the road are constantly changing, and their movement patterns are frequently unpredictable. The success rate of task offloading is decreased by the dynamic nature of vehicles. Before a task is finished, the vehicle requests that it be carried out and then shifts into a different position. Now the server must use sophisticated methods and algorithms to find the vehicle. Changes in the location of a vehicle entering or leaving a cluster of vehicles can have a negative impact on resource utilization and the computing paradigm's performance. To ensure the effectiveness and efficiency of mobile vehicles, the research community should therefore concentrate on vehicle mobility.

(2) Connectivity

A vehicle can be connected to another vehicle or a distant server through connectivity. A reliable internet connection must be present from the time the task is offloaded until the results are gathered in order to use the cloud server's services. Losing connectivity during this time period could lead to data loss, which could lead to the failure of the entire process. For MEC to offload and get the results of task execution, it needs a cellular connection. In the case of VEC, a connection to the central controller is always necessary. Task offload and result collection failures could be caused by a broken connection or no connection at all. Therefore, one of the unresolved problems for all computing paradigms is the connectivity issue.

(3) Task Migration

Vehicle users must offload compute-intensive and latency-sensitive tasks to edge servers due to capacity restrictions. Optimizing task migration choices is important because the channel environment is always changing and the topology is always shifting.

(4) Resource discovery and resource requests

Pay-as-you-go services are offered to users via the cloud. In vehicle edge computing, volunteer vehicles with sufficient extra resources provide these mission computing services for a fee or for free. In most cases, users are unaware of the nearby volunteer capacity. Therefore, it is an interesting and crucial area of research to develop a method that can accurately predict or indicate the presence of volunteer vehicles willing to display their computing resources along a route.

(5) Security and privacy

In a VEC, vehicles may not be fully trusted with each other due to dynamic topology changes. Also, there are security and privacy issues when different vehicle users are allowed to access the same physical edge server without strong protection mechanisms.

5. SUMMARY

The in-vehicle network and MEC are seamlessly integrated by VEC, bringing a wealth of resources to the vehicle while extending cloud capabilities to the network edge. VEC is an interesting new concept that offers low-latency, high-bandwidth, and highly reliable computing services to cars and other customers with limited computing resources. This is done to meet the growing need for computing and storage resources for in-vehicle applications. This paper provides a thorough background and explanation of task offloading in a VEC environment. The VEC architecture, benefits, and alluring application scenarios are then discussed. Then, through classification, we present a thorough literature review of prior work on game-theoretic offloading mechanisms for vehicular edge computing. Finally, future problems in research are discussed. Numerous issues need to be addressed as VEC research is still in its early stages. Intelligent transportation systems that are supported by vehicle edge computing are anticipated to make significant contributions to the digital transformation and technological advancement of the road transportation sector.

ACKNOWLEDGMENTS

This work is supported by the Qinghai Province Basic Research Project (2023-ZJ-713), Hebei Provincial Key Research Project (No.19270318D), Hebei IoT monitoring Engineering Technology Research Center (3142018055), Qinghai Key Lab of IoT (2020- ZJ-Y21).

REFERENCES

[1] Zhang, H., Lu, X., Vehicle communication network in intelligent transportation system based on Internet of Things[J]. Computer Communications, 160: 799-806 (2020).
[2] Alabbasi, A., Aggarwal, V., Joint information freshness and completion time optimization for vehicular networks [J]. IEEE Transactions on Services Computing, 15(2):1118-1129 (2020).
[3] Boukerche, A., Soto, V., Computation offloading and retrieval for vehicular edge computing[J]. Acm Computing Surveys, 6 (2020).
[4] Liu, Y., Wang, S., Zhao, Q., et al., Dependency-aware task scheduling in vehicular edge computing[J]. IEEE Internet of Things Journal, 7(6): 4961-4971 (2020).
[5] Hindia, M. N., Reza, A. W., Dakkak, O., Awang Nor, S., Noordin, K. A., Cloud computing applications and platforms: A Survey, In: Proceedings of the 3rd International Conference on Computer Engineering & Mathematical Sciences (ICCEMS 2014), pp.515-524 (2014).
[6] Abdelghany, S. Y., Hassan, H. M., Get as you pay model for IaaS cloud computing[C]. 2018 International Conference on Smart Communications and Networking (SmartNets), pp.1-6 (2018).
[7] Ahmad, F., Kazim, M., Adnane, A., Awad, A., Vehicular cloud networks: architecture, applications and security issues[C]. 2015 IEEE/ACM 8th International Conference on Utility and Cloud Computing (UCC), pp.571-576 (2015).
[8] Chen, X., Jiao, L., Li, W., et al., Efficient multi-user computation offloading for mobile-edge cloud computing[J]. IEEE/ACM transactions on networking, 24(5): 2795-2808 (2015).

[9] Shi, W., Cao, J., Zhang, Q., et al., Edge computing: vision and challenges[J]. IEEE internet of things journal, 3(5): 637-646 (2016).
[10] Khan, L. U., Yaqoob, I., Tran, N. H., et al., Edge-computing-enabled smart cities: A comprehensive survey[J]. IEEE Internet of Things Journal, 7(10): 10200-10232 (2020).
[11] Abbas, N., Zhang, Y., Taherkordi, A., et al., Mobile edge computing: a survey[J]. IEEE Internet of Things Journal, 5(1): 450-465 (2017).
[12] Zhang, Q., Gui, L., Hou, F., et al., Dynamic task offloading and resource allocation for mobile-edge computing in dense cloud RAN [J]. IEEE Internet of Things Journal, 7(4): 3282-3299 (2020).
[13] Raza, S., Wang, S., Ahmed, M., et al., A survey on vehicular edge computing: architecture, applications, technical issues, and future directions[J]. Wireless Communications and Mobile Computing (2019).
[14] Ye, D., Wu, M., Kang, J., et al., Optimized workload allocation in vehicular edge computing: a sequential game approach[C]. International Conference on Communicatins and Networking in China, pp.542-551 (2017).
[15] Jang, I., Choo, S., Kim, M., et al., The software-defined vehicular cloud: a new level of sharing the road[J]. ieee vehicular technology magazine, 12(2): 78-88 (2017).
[16] Abdelhamid, S., Hassanein, H. S., Takahara, G., Vehicle as a resource (VaaR) [J]. IEEE Network, 29(1):12-17 (2015).
[17] Xie, R. C., Lian, X. F., Jia, Q. M., et al., A review of offloading techniques for mobile edge computing [J]. Journal on Communications (2018).
[18] Wang, J., Pan, J., Esposito, F., et al., Edge cloud offloading algorithms: Issues, methods, and perspectives [J]. ACM Computing Surveys (CSUR), 52(1): 1-23 (2019).
[19] Mach, P., Becvar, Z., Mobile edge computing: a survey on architecture and computation offloading[J]. IEEE communications surveys & tutorials, 19(3): 1628-1656 (2017).
[20] Gu, X. H., Zhang, G. A., A review of mobile edge computing in in-vehicle networks[J]. Computer Application Research, 37(6): 1615-1621 (2020).
[21] Yuan, Q., Zhou, H., Li, J., et al., Toward efficient content delivery for automated driving services: an edge computing solution[J]. IEEE Network, 32(1): 80-86 (2018).
[22] Chen, Z., He, Q., Mao, Z., et al., A study on the characteristics of douyin short videos and implications for edge caching[C]. Proceedings of the ACM Turing Celebration Conference-China, pp.1-6 (2019).
[23] Chen, C., Qiu, T., Hu, J., et al., A congestion avoidance game for information exchange on intersections in heterogeneous vehicular networks [J]. Journal of Network and Computer Applications, 85: 116-126 (2017).
[24] Verma, T., Kumar, A., Fuzzy solution concepts for non-cooperative games [M]. New York, NY: Springer International Publishing (2020).
[25] Han, Z., Niyato, D., Saad, W., Başar, T., Hjørungnes, A., Game theory in wireless and communication networks: theory, models, and applications [M]. Cambridge university press (2012).
[26] Huang, X., Yu, R., Liu, J., et al., Parked vehicle edge computing: Exploiting opportunistic resources for distributed mobile applications [J]. IEEE Access, 6: 66649-66663 (2018).
[27] Klaimi, J., Senouci, S. M., Messous, M. A., Theoretical game approach for mobile users resource management in a vehicular fog computing environment[C]. 2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC), pp.452-457 (2018).
[28] Zhang, K., Mao, Y., Leng, S., et al., Optimal delay constrained offloading for vehicular edge computing networks[C]. 2017 IEEE International Conference on Communications (ICC), pp.1-6 (2017).
[29] Luong, N. C., Wang, P., Niyato, D., et al., Resource management in cloud networking using economic analysis and pricing models: a survey [J]. IEEE Communications Surveys & Tutorials, 19(2): 954-1001 (2017).
[30] Zeng, F., Chen, Q., Meng, L., et al., Volunteer assisted collaborative offloading and resource allocation in vehicular edge computing [J]. IEEE Transactions on Intelligent Transportation Systems, 22(6): 3247-3257 (2020).
[31] Li, G. S., Zhang, Y., Wang, M. L., et al., Resource management framework based on the Stackelberg game in vehicular edge computing [J]. Complexity (2020).
[32] Zhang, H., Wang, Z., Liu, K., V2X offloading and resource allocation in SDN-assisted MEC-based vehicular networks[J]. China Communications, 17(5): 266-283 (2020).
[33] Pham, X. Q., Huynh-The, T., Huh, E. N., et al., Partial Computation Offloading in Parked Vehicle-Assisted Multi-Access Edge Computing: A Game-Theoretic Approach[J]. IEEE Transactions on Vehicular Technology, 71(9): 10220-10225 (2022).

[34] Zhang, X., Peng, M., Yan, S., et al., Joint Communication and Computation Resource Allocation in Fog-Based Vehicular Networks [J]. IEEE Internet of Things Journal, 9(15), 13195-13208 (2022).
[35] Ritzberger, K., Foundations of non-cooperative game theory [J]. OUP Catalogue (2002).
[36] Liu, L., Wen, X., Wang, L., et al., Incentive-aware recruitment of intelligent vehicles for edge-assisted mobile crowdsensing [J]. IEEE Transactions on Vehicular Technology, 69(10): 12085-12097 (2020).
[37] Chen, C., Hu, J., Qiu, T., et al., CVCG: Cooperative V2V-aided transmission scheme based on coalitional game for popular content distribution in vehicular ad-hoc networks[J]. IEEE Transactions on Mobile Computing, 18(12): 2811-2828 (2018).
[38] Zou, L. S., Huang, X. W., Yang, J. M., et al., A review of resource allocation and pricing methods in mobile edge computing [J]. Telecommunications Science, 38(3): 113-132 (2022).
[39] Lu, J., Li, J., Yuan, Q., et al., A multi-vehicle cooperative routing method based on evolutionary game theory[C]. 2019 IEEE Intelligent Transportation Systems Conference (ITSC), pp. 987-994 (2019).
[40] Misra, S., Wolfinger, B. E., Achuthananda, M., et al., Auction-based optimal task offloading in mobile cloud computing [J]. IEEE Systems Journal, 13(3): 2978-2985 (2019).
[41] Wang, S., Zhang, Z., Yu, R., et al., Low-latency caching with auction game in vehicular edge computing[C]. 2017 IEEE/CIC International Conference on Communications in China (ICCC), pp.1-6 (2017).
[42] Hui, Y., Cheng, N., Su, Z., et al., Secure and personalized edge computing services in 6G heterogeneous vehicular networks[J]. IEEE Internet of Things Journal, 9(8): 5920-5931 (2021).
[43] Zhang, J., Lou, W., Sun, H., et al., Truthful auction mechanisms for resource allocation in the Internet of Vehicles with public blockchain networks [J]. Future Generation Computer Systems, 132: 11-24 (2022).