Engineering Science and Technology, an International Journal 54 (2024) 101699

Contents lists available at ScienceDirect

Engineering Science and Technology, an International Journal

journal homepage: www.elsevier.com/locate/jestch

Review

Task offloading for edge-IoV networks in the industry 4.0 era and beyond: A high-level view

Marieh Talebkhah [a,*], Aduwati Sali [a,b], Vahid Khodamoradi [a], Touraj Khodadadi [c], Meisam Gordan [d]

[a] Wireless and Photonics Networks Research Centre (WiPNET), Department of Computer and Communication Systems Engineering, Faculty of Engineering, Universiti Putra Malaysia (UPM), 43400 Serdang, Selangor, Malaysia
[b] Institute for Mathematical Research (INSPEM), Universiti Putra Malaysia (UPM), 43400 Serdang, Selangor, Malaysia
[c] Department of Information Technology, Malaysia University of Science and Technology, 47810 Petaling Jaya, Selangor, Malaysia
[d] School of Civil Engineering, University College Dublin (UCD), Belfield, D04 V1W8 Dublin, Ireland

A R T I C L E I N F O

Keywords:
5G
6G
Transport Infrastructure
Smart Internet of Vehicles (IoV)
Edge Computing (EC)
Offloading
Task Offloading
Internet of things (IoT)
Industry 4.0

A B S T R A C T

As a promising platform on the Internet of Things (IoT), the smart Internet of Vehicle (IoV) has emerged with the advent of the key connectivity to Industry 4.0, i.e. Fifth-Generation Mobile Communication (5G). However, problems with adequate battery life, powerful computing, and energy economy have hampered the development of this technology in light of the enormous increase in data traffic in 5G and 6G mobile communication networks. To address these limitations, this study proposes an Internet of Vehicles (IoV) system empowered by Edge Computing (EC), wherein intelligent vehicle nodes interact with an anchor node integrated with an EC server for data upload and download. Rather than solely focusing on enhancing the central cloud infrastructure, the integration of EC and IoT enables real-time and efficient services, thereby bolstering the storage and processing capabilities of underlying networks. By employing an offloading strategy within the Edge Computing-based Internet of Vehicles (EC-IoV) framework, users can allocate their workloads to suitable EC servers, leading to improved resource management and computational capabilities. However, challenges persist in evaluating the impact of uncertain user-EC server connectivity on offloading decision-making and mitigating potential declines in offloading efficiency.

1. Introduction

With the continuous technological advancements in our modern cities over the past decades, infrastructure management has encountered growing challenges, necessitating the adoption of novel and efficient methods for monitoring and maintaining of transportation infrastructure (e.g. airports, bridges, tunnels, roadways, and ports [1,2]. In this direction, the main aim of Intelligent Transportation Systems (ITS) is to enhance transportation mobility and safety, as well as improve the integration of advanced technologies into the transportation infrastructure [3]. Vehicular Ad Hoc Network (VANET) [4] plays a significant role as a facilitator in ITS. Being a special kind of Mobile Ad Hoc Networks (MANETs), VANETs are comprised of two basic elements: vehicles and Road-Side Units (RSUs) [5,6]. Vehicles are equipped with communication devices, which enables short-range wireless transportation. RSUs are distributed along the road to be connected to the backbone network for the purpose of facilitating network access. Data communication in VANETs can be realized in two models: Vehicle-to-Vehicle (V2V) and Vehicle-to-RSU (V2R) [7,8]. Using the two communication models, vehicular networks support an array of applications, which include three main categories: 1) road safety applications (e.g., lowering the risk of accidents); 2) traffic efficiency applications (e.g., reducing travel time and alleviating traffic congestion; and 3) value-added applications (e.g., providing infotainment, path planning and internet access).

The rapid evolution of vehicular networks is poised to facilitate the widespread adoption of smart vehicles, enabling a diverse array of applications [9,10]. However, the implementation of these applications requires substantial resources for data storage and processing. Due to the limitations in computation and communication capacities of vehicles, meeting the increasing resource demands, especially for applications with intensive computation and stringent delay requirements, poses a

* Corresponding author.
E-mail addresses: marieh.talebkhah@yahoo.com, marieh.t2018@gmail.com (M. Talebkhah), aduwati@upm.edu.my (A. Sali).

https://doi.org/10.1016/j.jestch.2024.101699
Received 12 October 2023; Received in revised form 8 April 2024; Accepted 21 April 2024
Available online 9 May 2024

challenge. To address these issues, Mobile Cloud Computing (MCC) has emerged as a widely recognized and promising solution [11]. By integrating computation and communication technologies, MCC allows the execution of user application services on remote cloud infrastructure. Consequently, MCC offers users several advantages, including: 1) reduced energy consumption; 2) the ability to support sophisticated services; and 3) access to substantial storage capacity. Various survey articles have explored MCC from different perspectives [12–15]. In [13], the definition, architecture, and application of MCC were introduced, along with an overview of existing challenges and corresponding approaches. The work presented in [16] delved into applications, challenges, and opportunities associated with MCC. Dedicated efforts have been directed towards the integration of MCC with vehicular networks to enhance road safety and elevate travel comfort [12]. In [17], the authors introduced a cloud-supported gateway model designed to enable seamless internet access in ITS, ultimately enhancing the overall user experience. The proposed model contributes to improved connectivity. Additionally, a VANET-CC model was introduced in [18], leveraging Cloud Computing (CC) resources to enhance the Quality of Service (QoS). The cloud system presented in [19] facilitated vehicles in locating their requested resources through mobile services.

Despite the benefits of MCC, the considerable distance between the cloud and users leads to high transmission latency. Additionally, the exponential growth of mobile data poses a substantial burden on the load of backhaul networks. Sending all data to the cloud for processing results in significant bandwidth consumption and competition. To address these challenges, Mobile Edge Computing (MEC), also interchangeably referred to as Fog Computing (FC), is envisioned as a promising paradigm [20–23]. In MEC, cloud services are brought to the network edge, meaning that computation and storage resources are relocated in proximity to users. This approach significantly reduces latency and contributes to substantial energy savings. Several survey papers in the MEC literature provide comprehensive overviews [24,25]. The research presented in [25] offers a comprehensive overview of current developments in MEC, covering advantages, architectures, and applications. Additionally, the paper delves into the issues surrounding

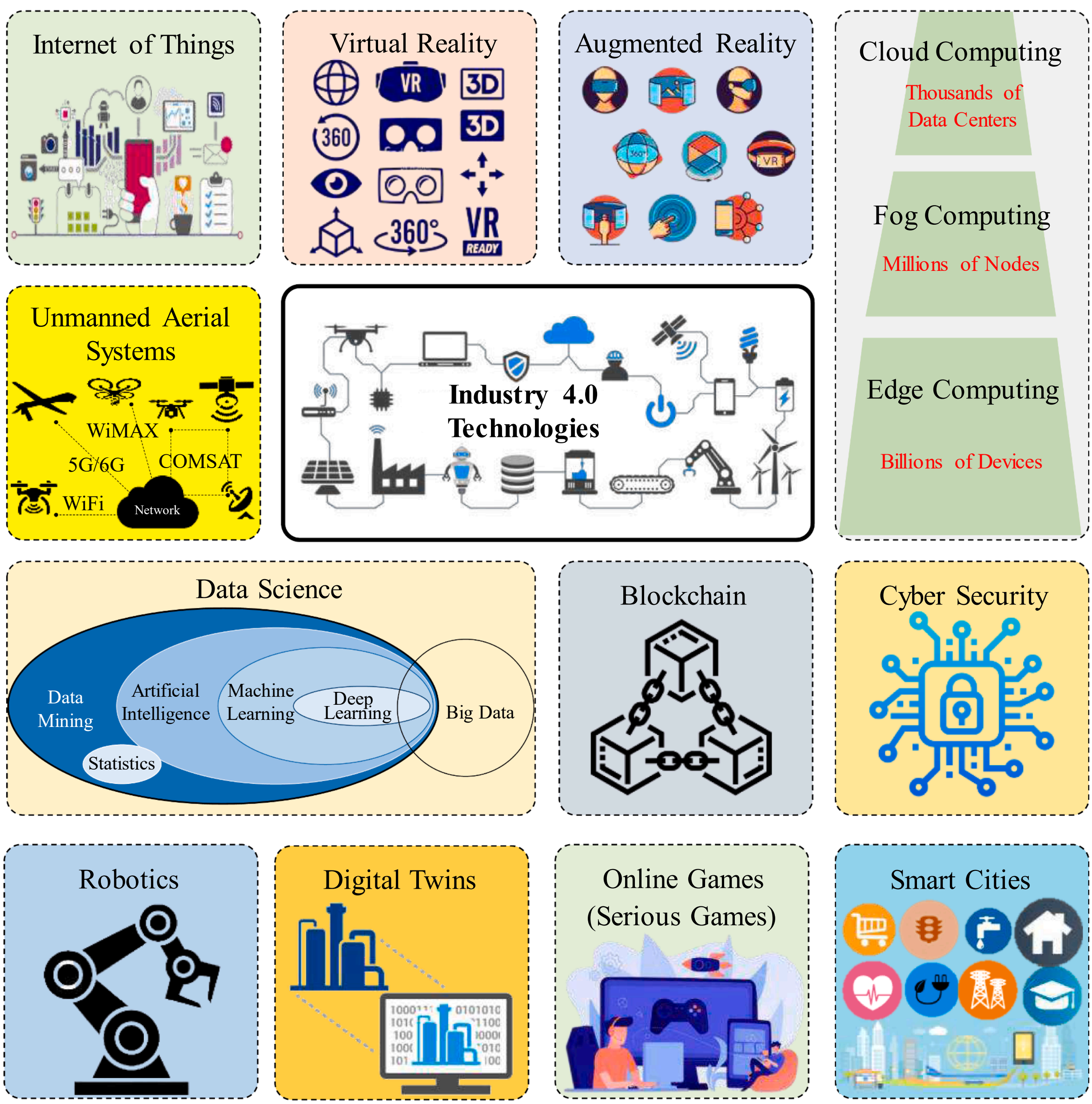

Fig. 1. Industry 4.0 technologies.

security and privacy, providing insights into existing solutions. In [26], the authors introduced work on computing and communication in MEC, with a specific focus on joint radio and computational resource allocation. Enabling technologies in MEC, including Virtual Machine (VM), Software Defined Networking (SDN), and Network Function Virtualization (NFV), are thoroughly discussed in [27].

Regarding the rapid expansion of IoT, the storage and processing capabilities of these IoT devices must be inevitably integrated to offer real-time and prompt services [28,29] recent years have witnessed the emergence of massive Industry 4.0 data computing and the demand for advanced services that provide an enhanced user experience like automatic navigation, unmanned driving, virtual reality, augmented reality, ultrahigh definition videos, and online games (see Fig. 1). For example, Fig. 1 illustrates the widespread utilization of Unmanned Aerial Vehicles (UAVs) and Internet of Drone (IoD) resources in delivering various anticipated services and applications [30,31]. By incorporating fog servers into UAVs, it leverages the advantages of both FC and drone technologies to offer unprecedented benefits such as coverage expansion, low latency, and flexible processing with the trade-off of increased energy usage [32]. These fog-enabled drones can be deployed rapidly to specific locations, fly directly to where data is generated, and provide on-demand processing capabilities to nearby IoT devices. As a result, they empower IoT networks with enhanced real-time data analytics, low-latency responses, and improved data privacy, all while ensuring a higher degree of fault tolerance [33,34].

Issues such as communication latency and expensive operations have seriously challenged the use of CC-based remote computing task loading mode. EC is often integrated with IoT. EC is capable of supplying the services and CC demands of wireless users [21]. As a consequence, it can provide the underlying networks with real-time and low-latency services [35]. EC requires less number of data centers in comparison with traditional cloud architectures, making it a promising candidate to meet the advanced services' demands for responsiveness, latency, and privacy. The current trend is to develop the EC by integrating wireless technology and mobile computing [36–38]. However, there is a growing need for more practical approaches to reduce the time it takes for mobile terminals and networks to respond and minimize the amount of energy they consume, in order to achieve better resource management.

IoV requires a collaboration between the vehicles and infrastructure for the delivery of value-added ITS services like infotainment services, traffic management, accident reduction, and route recommendation [29]. The IoV establishes a connection between the ITS devices and CC servers, i.e. The place for processing and analytics [21]. Nonetheless, the offload of a great deal of data from geologically distributed smart devices and vehicles may lead to network overhead and bottlenecks, requiring excessive network resources. Furthermore, the application of the remote cloud servers for the analysis of the ITS data streams leads to long processing and response times, which may not be tolerated by the latency-sensitive ITS applications. IoV is a novel paradigm for enhancing vehicle-user information interactions to improve urban traffic [36]. In an IoV environment, the vehicles are connected to various transmitters and receivers can transport required signals to connect vehicles to remote infrastructures or other vehicles [37]. The discrepancy between the capacity limitations and the communication service needs of the vehicles is a serious problem considering the fast increment of road traffic. The onboard network, on the other hand, is challenged by the requirement of ubiquitous connection and high-quality service for a large number of vehicles. The aforementioned issues can be addressed through the utilization of EC technology, which leverages the computational capabilities at the edge of vehicle wireless access networks [38,39]. IoV is an application of the IoT technology in the area of intelligent transportation capable of intelligent management of the traffic and offering more mature applications in terms of path planning, navigation, online interactive games, autonomous driving, augmented reality, intelligent-assisted driving, and other media applications for passengers [40,41]. The lightweight edge server on the roadside unit, however, does not suffice to handle various computational tasks with diverse granularity and QoS requirements. Therefore, providing an efficient operation for complex services specifically, supplying dependable connectivity and top-notch network services for numerous vehicles could be a serious challenge.

The paradigm of CC can offers computing resources on a "pay-as-you-go" basis [29] which can be easily accessed through the Internet from any place at any time [21]. CC has been recently employed in data storage, processing, and analysis for the IoV. At the same time, some applications related to vehicular networking were deployed to the cloud to supply relevant services to customers. The cloud load is increasing with the exponential rise in the number of vehicles and mobile terminals. Furthermore, the relatively long distance between the CC centers and end users can result in great processing latency, posing serious challenges to the latency-sensitive applications in the IoV, like an ambulance requiring its surrounding traffic data in real time for timely arrival at the rescue site, or a moving vehicle in need for instantaneous information to warn collision.

To transmit tasks and handle EC resources in IoV, EC can act as a core access point [42]. The centralized cloud models have limitations when it comes to the average task latency and resource cost, primarily due to the inherent delay caused by the distance between the edge IoT device and data center [43,44]. Regarding the exponential growth of edge devices, applications requiring end-to-end communication may face major difficulties due to the high latency of these devices.

Vehicular Edge Computing (VEC) is an innovative networking paradigm that aims to enhance the computational capabilities of vehicular networks. The increasing demand for modern vehicular applications has presented a challenge in meeting the communication and computational requirements. With VEC, service providers can host services near smart vehicles, resulting in reduced latency and improved QoS. Unlike centralized services like Vehicular Cloud Computing (VCC), VEC is designed for applications with distributed deployments. By extending the benefits of centralized cloud services to the network edge, VEC offers several advantages [40,41,45]. EC more efficiently improves the computational capacity in vehicular environments [46]. In EC, data processing and analysis are performed near the end devices, with the edge serving as an intermediary between the cloud and vehicles. Edge nodes, which are servers with ample computational and storage capacities, are deployed near vehicular networks. This proximity enables EC to offer improved QoS by providing computing and storage services in close proximity to the users. Moreover, to support modern applications within vehicular networks, a robust communication and computational mechanism is necessary [42].

In the IoV, a plethora of vehicle-network services has emerged, including traffic jam notification and danger alarming services [47]. Additionally, a growing number of vehicles are now equipped with multimedia devices to offer entertainment services to passengers within the vehicles [48]. These services result in significant data flow, and the generated data is intended for sharing among users [49,50] Consequently, these services need to be meticulously designed in alignment with the tasks' requirements and the capabilities of Edge Computing Devices (ECDs). Therefore, effective resource management is crucial for efficiently offloading computing tasks in the VEC system.

The sensors in vehicles gather data, which is then processed and stored by the edge servers. These services enable low-latency communication with increased context awareness. EC offers numerous advantages for low-latency applications, including safety applications such as driving safety and context awareness, as well as non-safety applications like video streaming, Augmented Reality (AR), and infotainment. VCC is compared with VEC in Table. 1.

1.1. Motivation and contribution

One of the primary challenges in VEC is the Computing Offloading (ComOf) process, which involves vehicles selecting the optimal edge

Table 1
Comparison between VEC and VCC.

Features	Vehicle Edge Computing (VEC)	Vehicle Cloud Computing (VCC)
Location	At user's proximity	Remote Location
Latency	Low	High
Mobile Support	High	Limited
Decision Making	Local	Remote
Communication	Real Time	Constraints in Bandwidth
Security	High	Limited
Reliability	High	High
Architecture Scability	High distributed	Limited centralized
Storage Capacity	Limited	Highly Scalable
Context Awareness	Yes	No
Power Consumption	Limited	High
Platform	Mostly ASIC	Mostly CPU, GPU, FPGA
Device Heterogeneity	Highly Supported	Limited Supported
Computing Capability	Medium	High
Cost of Development	Low	High

nodes in real-time while considering criteria such as latency, cost, and Energy Efficiency (EE). Furthermore, it is important for service providers to generate revenue through such programs. Another critical issue is the caching of content at specific edge nodes and delivering it directly to the relevant vehicles. Uncertainty still exists regarding the methods for data computation offloading, which are closely tied to the optimization problem of VEC resource management. According to the literature, ComOf focuses mostly on optimizing scheduling and task allocation processes, which is the main focus of our study. ComOf allows for the execution of computationally demanding and time-sensitive activities in an edge server while also reducing processing delay and energy usage. The foundation of VEC is ComOf. However, because of the high mobility and changing network topology of vehicles in IoV environment, computation offloading in VEC is fraught with difficulty. Furthermore, each vehicular terminal in the VEC, in contrast to other commonly used mobile terminals, can serve as both a task vehicle for task execution and a task vehicle for service requests simultaneously.

The studies described in Table. 2, which provides a summary of the existing research in the field, highlight that while many studies address task offloading in vehicle contexts, there is a noticeable gap in discussing task offloading in vehicular environments from the viewpoint of the vehicular communication network.

In general, existing surveys have covered various topics such as MEC [39], opportunistic offloading [40], mobile data offloading methods [41,45], particularly in cellular networks [46], and game theory in multi-access edge computing [42]. However, only a few studies have specifically focused on VEC [43].

As a result, ComOf in VEC is a crucial area for research. We examine and provide a summary for computation offloading on VEC after doing a thorough investigation and research on VEC. The contribution of our survey in this area is extensive. Our survey offers a concise yet comprehensive overview of the VEC concept. We delve into its intricacies, exploring its architecture, layers, communication technologies, and diverse range of vehicular applications. By doing so, we establish a solid foundation for understanding the critical aspects that shape computational offloading challenges in VEC.

In conclusion, our survey not only identifies the existing research gaps and unsolved challenges within the VEC domain but also sheds light on potential future research directions. By highlighting these opportunities, we aim to inspire both novice and experienced researchers to delve deeper into this exciting field. We firmly believe that this survey will serve as a valuable resource, providing a solid foundation for further advancements and significant contributions in the realm of VEC.

Our survey goes beyond a mere literature review and provides novel

Table 2
Overview of related works.

Reference	Task Offloading V2V	V2I	V2X	Main Contribution
[40]	×	×	×	Exploring voluntary opportunity offloading Techniques considering traffic and computational offloading protocols. However, this survey primarily relied on mobile device-based data and centralized task offloading.
[51]	×	×	×	In this paper, the authors initially highlighted a Software-Defined Vehicular Edge Computing (SD-VEC) architecture. In this architecture, a controller plays a dual role by guiding the strategy for task offloading from vehicles and also determining the strategy for allocating edge cloud resources. To derive the optimal strategies, they formulated a problem related to the selection of edge clouds and the allocation of resources. The objective of this problem is to maximize the likelihood that a task will be successfully completed within a predefined time limit.
[52]	×	×	×	The researchers designed a dynamic approach to offload specific components or modules of vehicular applications. They created heuristic mechanisms for the placement and scheduling of these modules, considering the on-board unit versus the cloud. Notably, their design's key feature is its capability to flexibly offload computations to the cloud, making dynamic decisions based on varying network conditions.
[53]	×	×	×	This paper presented a systematic literature review of ComOf schemes and methods within the domain of VEC. It categorized the existing research on ComOf into distinct categories.
[54]	×	×	×	In this paper, a literature review is developed to explore the concept of computation offloading in EC. Various facets of computation offloading, such as energy consumption minimization, QoS, and Quality of Experience (QoE), are thoroughly examined.
[55]	×	×	×	This survey is examined to comprehensively review and structure the existing body of literature on computation offloading within vehicular environments. Furthermore, it aims to clarify certain concepts, introduce a taxonomy highlighting critical aspects, and categorize the majority of works in this field based on their respective categories.
[56]	×	×	×	In this paper, Authors presented an overview of VEC, covering its introduction, architecture, key enablers, advantages, challenges, and various appealing application scenarios. Subsequently, they detailed several common research areas where VEC finds application.
[9]	×	×	×	In this manuscript, the authors delineated various facets of VEN, with a specific focus on VEC. This included an examination of its structural elements, hierarchical layers, communication mechanisms, as well as its roles in ComOf and content caching and delivery (CachDel) scenarios. Additionally, they conducted an appraisal of the current methodologies employed to address the challenges in ComOf and CachDel within the framework of VEC architecture. In conclusion, the authors underscored noteworthy obstacles, unresolved matters, and potential areas for future research in the domains of ComOf and CachDel within the context of VEC.
[57]	×	×	×	This study concentrated on the offloading of computational tasks within VEC. It surveyed the primary offloading schemes and methods within the VEC domain and categorized the

(*continued on next page*)

Table 2 (*continued*)

Reference	Task Offloading V2V	V2I	V2X	Main Contribution
				current offloading of computational tasks into distinct categories.
[58]	×	×	×	In this paper, the authors classified state-of-the-art computation resource allocation schemes using three key criteria: (1) Their optimization objectives, (2) The mathematical models/algorithms employed, and (3) The primary technologies applied. Additionally, they identified and discussed ongoing challenges related to computation resource allocation in VEC and proposed potential avenues for future research.
[59]	×	×	×	They offered an extensive overview of all computing paradigms associated with vehicular networks. Additionally, they presented the architectural specifics, commonalities, distinctions, and crucial attributes of each computing paradigm. The study concluded by highlighting outstanding research challenges within vehicular networks and suggesting potential directions for future research.
[60]	×	×	×	This study introduced an innovative federal classification distinguishing between cloud, edge, and fog computing. It also outlined a research roadmap for offloading in diverse federated scenarios. The authors conducted a comprehensive literature survey to explore the various optimization methods employed in addressing the offloading challenge, comparing their notable characteristics. Additionally, they presented a survey on offloading within federated systems, specifically focusing on machine learning approaches, and shared valuable insights gained from these surveys.
Our paper	✓	✓	✓	**We begin by establishing a foundation in vehicular communication technologies, communication modes, and various computing architectures. We comprehensively examined the architecture of VEC, exploring it across three distinct layers (Cloud, Edge, and Smart Vehicle layers). Finally, we address some open issues and outline future work in our paper.**

contributions in the form of brand-new summary tables and valuable insights gained from studying the task offloading domain. These summary tables serve as comprehensive references, presenting a consolidated view of the key findings and approaches in task offloading research. By distilling the essence of existing studies, our survey offers a valuable resource for researchers and practitioners seeking a deeper understanding of task offloading in various domains.

In conclusion, we shed light on the remaining research challenges and identify promising future research directions in this dynamic and evolving field. By pinpointing the unresolved issues, we aim to inspire further investigations and stimulate the curiosity of researchers. We believe that addressing these open research problems will not only enhance our understanding of the subject but also pave the way for innovative solutions and advancements in the field of task offloading.

We are confident that our survey will provide valuable insights and benefits to researchers across all levels of expertise, from newcomers to seasoned professionals. By presenting a comprehensive overview of the VEC landscape and highlighting key research findings and trends, our survey serves as a valuable resource for researchers to deepen their understanding of the field. Furthermore, we believe that our survey can serve as a catalyst for inspiring further research and innovation, acting as a springboard for researchers to make significant contributions to the advancement of VEC and its related domains.

The paper is structured as follows. In Section 2, we provide an explanation of the related works. In Section 3, we delve into the VEC architecture, examining its key components and the overall framework. Section 4 explores the concept of smart vehicles and its essential elements, along with an in-depth analysis of the smart vehicle network and its diverse range of services and applications. We explain the concept of offloading in Section 5. In section 6, we examine the realm of vehicular task offloading, exploring its diverse categories: V2V, V2I and V2X schemes. We shed light on the intricacies of each approach, elucidating their significance and contributions within the vehicular computing landscape. Section 7 encompasses the discussion of task offloading in Dynamic Edge-IoV networks. In Section 8, we categorize the technical issues related to VEC, discussing topics such as computation offloading, resource management, and network connectivity. Section 9 not only highlights the challenges and potential roadblocks associated with VEC implementation, but also sheds light on the open gaps found in the selected literature, along with a discussion of future research work. Finally, in Section 10, we provide concluding remarks summarizing the key findings and insights from the paper. The overall organization of the paper is visualized in Fig. 2, which illustrates the flow and structure of the paper. Please take note that Table. 3 contains all the acronyms and abbreviations used throughout the study.

2. Related works

Recent years have witnessed the continuous evolution of Information and Communication Technologies (ICT), which has improved the processing and computation capacity of diverse applications. Concerning the IoV, advancements such as AR and self-driving applications [36] led to the expansion of vehicle communication mechanisms due to higher connection and intelligence [35]. Such advancements require remarkable computational and massive data generation [37]. Delay-sensitive applications (emergency help and natural disaster rescue) have to be processed in specified time constraints [38]. They also require sufficient vehicular computational and communication resources. MCC has been used by researchers to offload the necessary work using high-capacity servers on a distant cloud [44]. Resource limitations can be solved through task offloading, which allows a vehicle with limited resources to complete its compute duties in a vehicle with abundant resources nearby [61]. The vast transmission distance between the source cars and the cloud servers, despite benefits like decreased energy usage and greater storage capacity, may cause network congestion and latency [62]. As a newly emerged research area, MEC is aimed to decline the transmission distance and computational load of the cloud. Upon integration with conventional vehicular networks, MEC can lead to VEC to bring the computational resources of a cloud closer to the end-user (vehicle). VEC has a major contribution to supplying edge services at the shorter delay and wider bandwidth [63]. RSUs, which are edge servers placed closer to the cars for real-time data collecting, processing, and storage, can be thought of in the context of VEC as suppliers of communication, compute, and storage. If the computational resources of the source vehicle do not suffice, the vehicle can offload the tasks to RSUs [64].

VEC has become the major trend and numerous studies have been devoted to resolving its challenges [65,66]. In addition, the VEC has offered a versatile paradigm for decreasing the computational burden of vehicles and offering real-time responses to the task requests [67].

Wang et al. [68] proposed a cooperative data processing approach using a multilayer model consisting of user, access, and cloud layers. In this model, computing and transmission resources were allocated to each device, and a convex optimization problem was formulated to maximize the effectiveness of resource allocation strategies. Their work aimed to enhance the overall data processing efficiency by leveraging cooperation among different layers of the network architecture.

A virtualization technology was adopted by Zhang et al. [69] for online allocation of resources in a dense cloud wireless access network. To strike a balance between minimizing delay and reducing mean

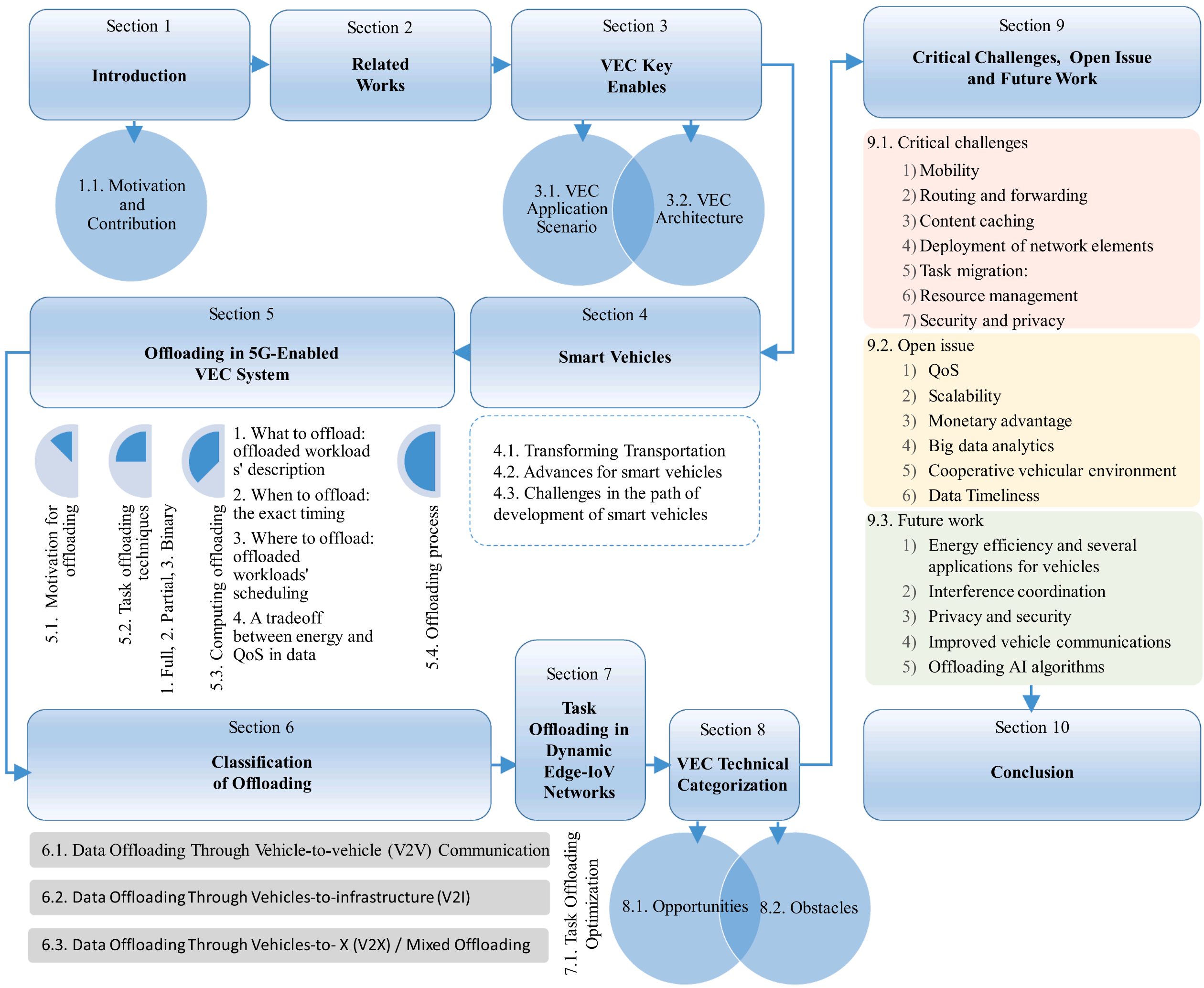

Fig. 2. Organization of the paper.

energy consumption, they employed the Lyapunov optimization theory. Additionally, Chen et al. [70] devised a search tree algorithm utilizing the branch and bound method to address the challenge of minimizing delay in computational offloading and resource allocation.

Zhao et al. [71] proposed a cloud-edge cooperation model to develop an optimal decision-making scheme for routing requests to either the edge server or the cloud for sequential processing. In this model, mobile device requests are transmitted through the access point in chronological order. For maximizing resource use, Ning et al. [72] combined several edge servers to offload computation and allocate cache. The utilization of EC significantly reduces the distance between on-board tasks and computing resources, enabling real-time services with minimal latency in the IoV cloud-EC compared to centralized cloud infrastructures. Kumar et al. [73] proposed an efficient and energy-saving resource scheduling strategy for IoV based on MEC. This strategy focuses on controlling energy consumption through edge servers, making it well-suited for large-scale and widely distributed vehicle networks.

Yu et al. [74] introduced an offloading approach based on MEC for IoV, aiming to identify the most suitable MEC server for task management. Their approach leveraged both computation and vehicle mobility factors in the offloading decision-making process. In the domain of multi-user fog computing, Zhang et al. [75] proposed an energy-saving ComOf strategy and developed a distributed algorithm using the alternating direction multiplier technique. Ma et al. [76] tackled the challenge of multi-user computational offloading in a multi-channel wireless interference environment by employing game theory techniques. They presented a computational offloading technique that optimizes resource allocation. Xiong and coworkers [77] proposed an optimization approach to enhance the distribution of computational and network resources, aiming to reduce transmission latency and computation time. distribution of computational and network resources to reduce transmission latency and computation time. Dai et al. [78] employed a real-time traffic management approach for vehicle offloading, leveraging fog computation to minimize the mean reaction time of vehicle computing tasks. They utilized queuing theory to develop a mathematical model for vehicle-based fog nodes, providing an initial solution to the offloading optimization problem. Zhou and colleagues [28] investigated a novel two-stage strategy for resource sharing and task offloading, integrating contract theory and computational intelligence. In the initial stage, they introduced an effective incentive mechanism, employing contract theory to encourage servers to share their remaining computational resources. The subsequent stage involved an analysis of a decentralized task offloading algorithm that leverages the online learning capabilities of a multi-armed bandit. Specifically, they addressed a distance-aware, occurrence-aware, and task-property-aware volatile upper confidence bound algorithm designed to minimize the prolonged delay in task offloading. To evaluate the effectiveness of the proposed algorithm, comprehensive simulations were conducted, confirming its performance.

Game theory has emerged as a powerful tool for analyzing and

Table 3
Abbreviations and acronyms.

Abbreviations and acronym	Description
5G	FIFTH-GENERATION WIRELESS COMMUNICATION
6G	SIXTH-GENERATION WIRELESS COMMUNICATION
AI	ARTIFICIAL INTELLIGENCE
AR	AUGMENTED REALITY
APS	ACCESS POINTS
BS	BASE STATION
BD	BIG DATA
CC	CLOUD COMPUTING
comof	COMPUTING OFFLOADING
CIOTS	CONSUMER INTERNET OF THINGS
CSOS	CONTEXT-SENSITIVE OFFLOADING SYSTEM
CAS	COLLISION AVOIDANCE SYSTEM
caas	COMPUTATION AS A SERVICE
CRL	CERTIFICATE REVOCATION LIST
CAS	COLLISION AVOIDANCE SYSTEM
DDPG	DEEP DETERMINISTIC POLICY GRADIENTS
DSRC	DEDICATED SHORT-RANGE COMMUNICATIONS
DRL	DEEP REINFORCEMENT LEARNING
EC	EDGE COMPUTING
EE	ENERGY EFFICIENCY
ECDS	EDGE COMPUTING DEVICES
EC-IOV	EDGE COMPUTING-BASED INTERNET OF VEHICLE
FC	FOG COMPUTING
GPS	GLOBAL POSITIONING SYSTEM
IOT	INTERNET OF THINGS
IOD	INTERNET OF DRONE
IOV	INTERNET OF VEHICLES
IGR	IMPROVED GEOGRAPHIC ROUTING
Iaas	INFOTAINMENT AS A SERVICE
ITS	INTELLIGENT TRANSPORTATION SYSTEM
ICT	INFORMATION AND COMMUNICATION TECHNOLOGIES
KNN	K-NEAREST NEIGHBORS
LTE	LONG-TERM EVOLUTION
LIDAR	LIGHT DETECTION AND RANGING
MAEC	MULTI ACCESS EDGE COMPUTING
MANETS	MOBILE AD HOC NETWORKS
MCC	MOBILE CLOUD COMPUTING
MEC	MOBILE EDGE COMPUTING
MBS	MICRO BASE STATION
MECO	MOBILE-EDGE COMPUTATION OFFLOADING
Naas	NETWORK AS A SERVICE
NFV	NETWORK FUNCTION VIRTUALIZATION
NVMe	NONVOLATILE MEMORY
OBCS	ON-BOARD COMPUTERS
OUE	ONBOARD USER EQUIPMENT
PSO	PARTICLE SWARM OPTIMIZATION
QOS	QUALITY OF SERVICE
QOE	QUALITY OF EXPERIENCE
RSUS	ROADSIDE UNITS
RADAR	RADIO DETECTION AND RANGING
saas	STORAGE AS A SERVICE
SDN	SOFTWARE DEFINED NETWORKING
SLAS	SERVICES LEVEL AGREEMENTS
UES	USER EQUIPMENT
UAVS	UNMANNED AERIAL VEHICLES
VANET	VEHICULAR AD HOC NETWORK
VCC	VEHICULAR CLOUD COMPUTING
VEC	VEHICULAR EDGE COMPUTING
V2V	VEHICLES-TO-VEHICLES
V2I	VEHICLE- TO- INFRASTRUCTURE
V2S	VEHICLE –TO –SENSOR
V2P	VEHICLE –TO- PEDESTRIAN
V2R	VEHICLE –TO –ROAD SIDE UNITS
V2N	VEHICLE –TO –NETWORK
V2E	VEHICLE- TO- EVERYTHING
VEN	VEHICLES EDGE NETWORK
VM	VIRTUAL MACHINE

optimizing resource allocation in IoT-Fog environments. In this context, IoT devices and fog nodes act as rational decision-makers aiming to maximize their own utility, often in the form of throughput, energy efficiency, or latency minimization. Game-theoretic models facilitate the understanding of interactions among these entities, considering factors such as competition for resources, cooperation incentives, and potential conflicts of interest [79]. Various game-theoretic frameworks, such as non-cooperative games, cooperative games, and evolutionary game theory, have been applied to address different aspects of resource allocation in IoT-Fog systems. These models enable the characterization of equilibrium solutions, such as Nash equilibrium or Pareto optimality, which provide insights into the stability and efficiency of resource allocation strategies in dynamic and heterogeneous environments [80]. Based on [80], the authors reviewed a computational framework that takes into account energy consumption and transmission latency as key factors in determining task offloading for IoT applications. They approached the problem by framing it as a game, wherein IoT devices aim to optimize task distribution to minimize energy consumption and latency collectively. They further devised a decentralized algorithm for task distribution, allowing devices to adapt their strategies based on the actions of others. Additionally, they demonstrated that their algorithm converges to a Nash equilibrium, ensuring stable outcomes. Lastly, the authors conducted thorough evaluations, comparing their computational model and findings with those from previous studies.

Liwang et al. [81] presented a 5G cloud-enabled scenario in VCC, where a vehicular terminal functions as either a service provider with available computation resources or a requestor with a computation-intensive task. This task can be executed locally or offloaded to nearby providers through opportunistic V2V communications. The study addressed three key issues: (i) determining the appropriate offloading rate for requestors; (ii) selecting the most suitable computation service provider; and (iii) identifying the optimal pricing strategy for each service provider. To address these challenges, they proposed a two-player Stackelberg-game-based opportunistic computation offloading scheme. This approach considers situations with both complete and incomplete information, focusing on factors such as task completion duration and service price. In the case of complete information, they simplified it into a common resource assignment problem with mathematical solutions. For incomplete information, they derived Stackelberg equilibria of the offloading game and discussed the corresponding existence conditions in detail. The effectiveness of the proposed methods is demonstrated through Monte-Carlo simulations, revealing a significant reduction in task completion duration. Simultaneously, the approaches ensure the profitability of service providers, leading to mutually satisfactory computation offloading decisions.

Swain and colleagues [82] reported an effective task offloading strategy named METO, which relies on matching theory principles to minimize total system energy consumption and the occurrence of task deadline violations in an IoT-Fog interconnected network. To address the multi-criteria nature of resource allocation, they established the weights of various criteria through the CRITIC method, considering inter-criteria correlations. Subsequently, to prioritize alternatives, they employed the TOPSIS approach. Utilizing this prioritization, they formulated the offloading problem as a one-to-many matching game and employed the Deferred Acceptance Algorithm (DAA) to achieve a stable assignment. They conducted simulations under two distinct scenarios involving the offloading of both homogeneous and heterogeneous tasks. Through extensive simulations in these environments, their proposed METO algorithm demonstrates superior performance compared to existing schemes, exhibiting enhancements in energy consumption, completion time, and execution time. Additionally, METO exhibits a reduced number of task deadline violations compared to the baseline methods used for comparison.

Chiti et al. [83] proposed an efficient strategy for offloading computationally intensive tasks from end-user devices to Fog Nodes. The computation offload problem was formulated as a matching game with externalities, with the aim of minimizing the worst-case service time by taking into account both computational and communications costs. In particular, this paper proposed a strategy based on the deferred acceptance algorithm to achieve efficient allocation in a distributed mode and ensure stability over the matching outcome. The performance of the proposed method was evaluated through computer simulations in terms of worst total completion time, mean waiting, and mean total completion time per task. Moreover, with the aim of highlighting the advantages of the proposed method, performance comparisons with different alternatives were also presented and critically discussed. Finally, a fairness analysis of the proposed allocation strategy was also provided on the basis of the evaluation of the Jain's index. In [84], the authors framed the issue of user-fog pairing as a matching game with minimum and maximum quota constraints. They introduced a Multi-Stage Deferred Acceptance (MSDA) algorithm to balance fog resource usage and improve response times for users. Simulation results demonstrated that the proposed model, when compared to a baseline user matching approach, resulted in reduced delays for users. Swain et al. [85] described a matching theory-based protocol, A-DAFTO, to address challenges in offloading computations from Consumer Internet of Things (CIoTs) to nearby fog nodes for real-time consumer applications. A-DAFTO distributed network and computational load while meeting application deadlines, utilizing the Artificial Cap Deferred Acceptance (ACDA) algorithm. Simulation experiments demonstrated A-DAFTO's effectiveness, achieving zero outages and a 15.32 % reduction in total offloading delay compared to baselines. Gue et al. [86] addressed this research to investigate the Mobile-Edge Computation Offloading (MECO) challenge in highly compact IoT networks and introduced a two-tier game-theoretic greedy offloading approach as a resolution. Substantial numerical findings affirm the outstanding effectiveness of implementing computation offloading across numerous edge servers in ultra-dense IoT networks.

IoT-fog networks leverage offloading to enhance data processing capabilities closer to the network edge, facilitating real-time decision-making and reducing latency [30]. Similarly, UAVs can leverage their mobility to overcome spatial constraints and enable flexible communication. However, their limited computing resources and battery power pose significant challenges for UAVs. In [87], a task offloading algorithm is introduced to aid UAVs in executing computationally demanding tasks. This algorithm offers two approaches for offloading tasks. The first approach, known as airborne offloading, enables UAVs to transfer their computational tasks to nearby UAVs equipped with sufficient computing and energy resources. The second approach, termed ground offloading, facilitates the transfer of tasks from a multi-level edge cloud unit, linked with a ground station, to an edge cloud server. In [88], researchers delve into a scenario involving a fleet of small UAVs engaged in an exploration mission. They furnish a thorough demonstration of the existence of Nash equilibrium and propose a distributed algorithm that converges to this equilibrium. Guo and Liu [89] introduced the integration of UAV-aided communication and MEC as a promising approach to address the increasing demands for Big Data (BD) processing in UAV-aided IoT applications. The proposed algorithm effectively reduced energy consumption during task execution. Additionally, in [90], a new UAV-enabled MEC system is proposed, facilitating interaction with IoT devices, UAVs, and ECs. To enhance quality of service, the authors devise an optimization problem aimed at minimizing the weighted sum of service delays for all IoT devices while considering UAV energy consumption.

3. VEC key enablers

3.1. VEC application scenario

The development of VEC has led to the emergence of a wide variety of applications (Fig. 3). Some of these scenarios will be discussed in the

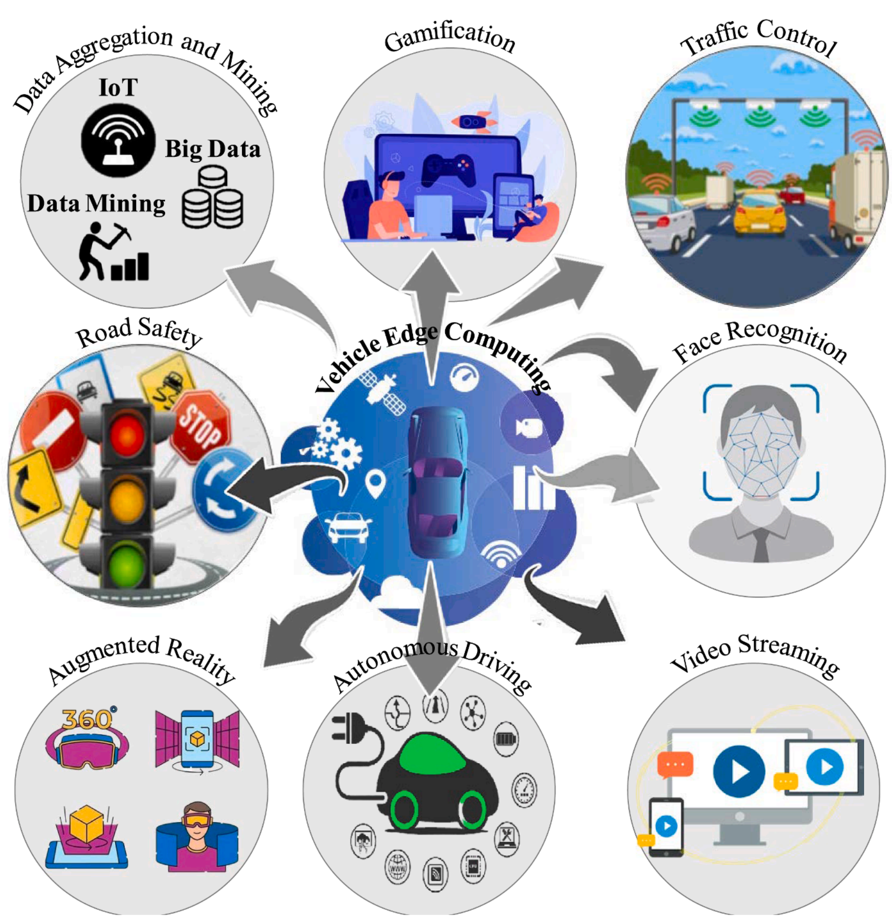

Fig. 3. Application scenarios of VEC.

following. Table. 4 exhibits a comprehensive overview of various applications, showcasing their specific bandwidth, latency, and data requirements. The table offers valuable insights into the distinctive characteristics of each application, including the volume of data they necessitate, their tolerance for time delay (latency), and the required bandwidth for efficient operation.

i. **Road Safety:** Vehicles and sensors installed along the road can continuously send data which can be real-time analyzed by the edge servers in the proximity of vehicles. Upon finding risk data, edge servers alarm the surrounding vehicles to avoid the hazard by taking proper actions, such as braking, changing the lane changing, or turning around. The Collision Avoidance System (CAS) [91] passes the collision-related data among the adjacent vehicles and warns the driver by beep sounds. As depicted in Fig. 4, to transmit information, this system makes use of the V2V network. As a collision mitigation system that uses radar and other sensors (working on laser or cameras to deal with the crash), CAS also functions as a precrash system to warn of impending collisions. It warns the driver on the cluster panel for on-time braking [4].
ii. **Entertainment:** By the emergence of smart vehicles, drivers are liberated from complicated driving tasks, thus, they have time to spend on entertainment, e.g., surfing the internet, playing gaming, or watching video [92]. The mentioned applications can use the advantages of the VEC computation and storage resources. For instance, drivers can directly fetch their desired videos without resorting to the remote cloud just through cooperative caching of popular contents among edge servers and vehicles. This will lower the delay and improve the experience of the user.
iii. **Traffic Control:** Edge servers cover a communication zone; any vehicle in this zone sends its current status (location, speed) and collected data (weather and road conditions) for the corresponding server. By analyzing the information from these vehicles, the server can have an insight into local traffic conditions to control the traffic flow and avoid traffic congestion [93].
iv. **Path Navigation:** Real-time navigation systems are of crucial significance in providing an optimal route. Real-time navigation requires data sensing, collection, and processing. Which can be supplied by the VEC.
v. **Ultra-low Latency Service:** VEC facilitates the delivery of services with ultra-low latency and high reliability, as highlighted in [94]. Applications like autonomous driving demand accurate and timely information about the surrounding environments, which can be effectively achieved through VEC by providing robust computational resources for executing critical tasks.
vi. **Computation-intensive Service:** VEC offers the potential to offload computation-intensive applications, such as AR and face recognition. Users of vehicles often face limitations in meeting the computation requirements of these applications due to limited resources. By leveraging the computation resources of VEC, these applications can be efficiently transferred to edge servers, enabling seamless execution, and enhancing the user experience.
vii. **Data Aggregation and Data Mining:** The increasing adoption of intelligent vehicles has resulted in a substantial rise in sensor-generated data and data exchanged among vehicles [95]. By deep exploitation of these data in VEC, more knowledge can be employed for improved data efficiency and network performance.

Table 4
Application requirements.

Application	Bandwidth	Delay	Source	Time
Health Monitoring System	High	Real Time	On-board Sensors	Sub-second
Infotainment	High	Real Time	On-board Sensors	Sub-second
Multi User Gaming	High	Real Time	On-board Sensors	Sub-second
Nearby Driver Collaboration	Low	Low	Nearby Vehicles	Several minutes
Platooning	Low	Real Time	Nearby Vehicles	Several minutes
Parking Lot's Information	Low	Low	Nearby Vehicles	Several minutes
Vehicles Tracking	Low	Low	Edge Coverage	Several hours
Traffic Light Management	Low	Low	Edge Coverage	Several hours
Emergency Vehicle Warning	Low	Low	Edge Coverage	Several minutes
File Sharing (Multimedia)	High	High	Entire network	Several days
Driver Behavior	Low	High	Entire network	Several days
Maps Update	High	High	Entire network	Several days

3.2. VEC architecture

The architecture, roles, and modes of operation are explained in this section for each component of the VEC system. According to Fig. 5, the VEC architecture is based on three layers: cloud, edge cloud, and smart vehicular layers.

i. **Cloud Layer:** The cloud layer is responsible for tasks that require computational capabilities beyond what the edge nodes can provide such as data mining, data aggregation, storage, batch processing, analysis optimization, and computation of complex data [96]. Furthermore, the cloud can compute a huge deal of data and complex computations rapidly. The infrastructure of the cloud includes two major parts: storage and computation. The collected data are sent to the cloud layer for permanent storage for future analyses. The computation part is responsible for computing and analyzing complex computational tasks in shorter times. Only non-latency-sensitive computing operations are transmitted through edge nodes to the cloud.
ii. **Edge Layer:** The edge layer serves as a crucial link between the smart vehicular layer and the cloud layer, ensuring a dependable connection. To achieve this, vehicles utilize wireless communication protocols (e.g. 802.11p, 3GPP, 3G, 4G, LTE, and 5G). The goal is to achieve low latency, location awareness, caching, content discovery, emergency management, and computation at a better QoS due to the proximity of the layer to vehicles, enabling real-time interactions [97]. Applications demanding rapid replies can be handled by the edge cloud layer with extremely low latency [98] such as environment recognition and video analytics. This layer offers the following services:

- Infotainment as a Service (IaaS): This service aims to enhance the user experience and passenger safety by providing a combination of information and entertainment options. The primary goal of IaaS is to provide passengers with a high-quality, interactive, and engaging experience during their journey. This service encompasses features such as real-time navigation, traffic updates, weather information, multimedia streaming, internet connectivity, and access to various entertainment options. By integrating information and entertainment functionalities, IaaS aims to make travel more enjoyable, convenient, and safe for the passengers.
- Network as a Service (NaaS): Users with internet connectivity can assist others by providing connection access through their vehicles, RSUs, or MBS. This service proves to be highly valuable, especially in

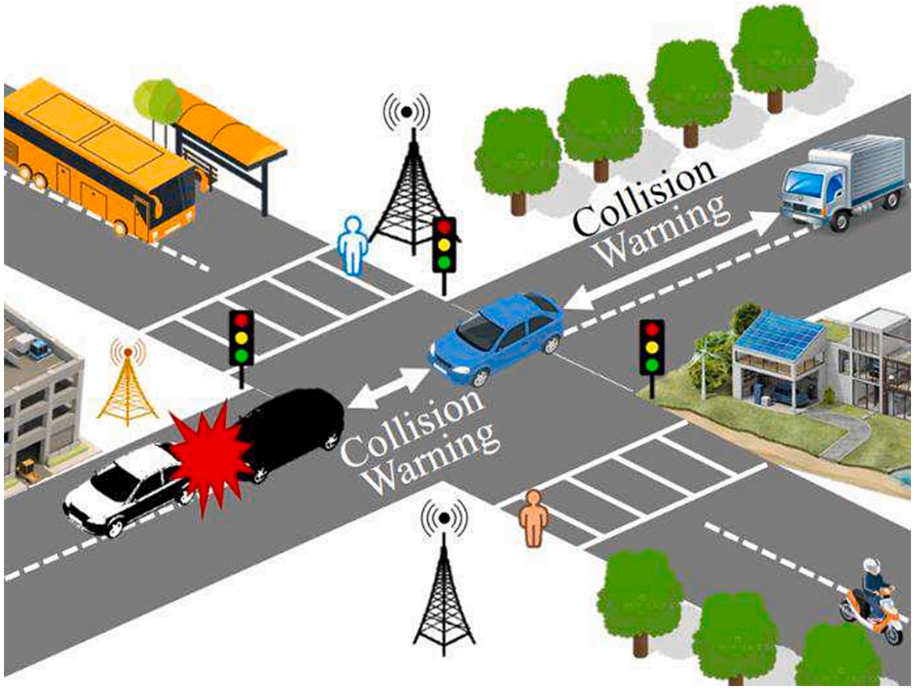

Fig. 4. Mechanism of CAS in IoV.

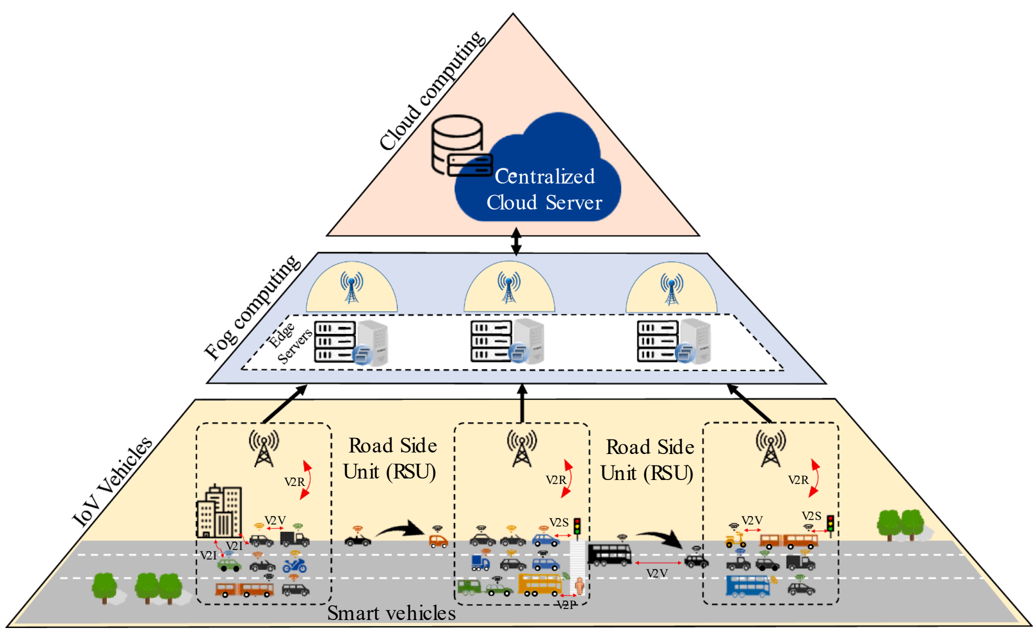

Fig. 5. Three-layer VEC architecture.

times of crisis. This service allows individuals to act as mobile network providers, extending network coverage and connectivity to areas where it may be limited or unavailable. NaaS becomes particularly valuable in emergency situations, as it can provide vital communication capabilities to affected areas where traditional network infrastructure may be disrupted. By leveraging the connectivity capabilities of vehicles and roadside units, NaaS offers an on-demand network service that can be deployed quickly and efficiently to assist in various scenarios, ensuring reliable communication and connectivity when it is most needed.

- Storage as a Service (SaaS): Vehicles may require additional storage to execute their programs or temporarily store backups. This need can be fulfilled by the edge server, which offers free storage services to cater to such requirements. SaaS allows vehicles to offload their storage requirements to the edge server eliminating the need for onboard storage expansion or relying solely on limited local storage capacity. By leveraging the storage resources provided by the edge server, vehicles can efficiently manage their data storage needs, ensuring seamless application operation and secure data backup.
- Computation as a Service (CaaS): Vehicles that are parked in parking lots or caught in traffic jams often have idle computational resources for extended periods. This presents an excellent opportunity for other vehicles or users who need to augment their processing capacity to handle heavy computational workloads.

Additionally, this layer serves as a platform for V2V communication and also facilitates communication between vehicles and external infrastructure (V2I). In the V2V scenario, vehicles interact with each other, allowing data to be transmitted through vehicles until it reaches the edge for further processing. Emergency alerts are broadcast to nearby vehicles as well as the edge if any vehicle exhibits unusual behavior (due to a change in direction, exceeding the posted speed limit, or mechanical failure). The location, speed, and direction of the vehicle may be included in this communication. Using infrastructures like roadside devices, micro base stations, and edge servers, the V2I offers a dependable platform for operational data exchange among vehicles through wireless networks.

iii. **Smart Vehicles Layer:** The smart vehicular layer encompasses a group of neighboring vehicles that communicate wirelessly to share computational and storage resources. This layer abstract information from various embedded sensors, GPS, cameras, radar, Lidar, and other devices presents in vehicles. The collected data can be transmitted to the edge cloud layer for storage or utilized by various application layer services. Consequently, this layer facilitates the observation of occupants' and drivers' behavior as well as the surrounding environment. The fundamental component of this layer is the vehicle itself, and in this paradigm, a smart car is one that is equipped with the latest sensors and communication technology. In the upcoming section, we will delve deeper into the concept of "smart vehicles", exploring their essential components, communication methods, services, and applications.

4. Smart vehicles

Any kind of vehicle that is capable of being autonomously driven and outfitted with OBUs, several sensors, including RADAR, LIDAR, GPS, videography, and cameras are developed to enhance the movement of vehicles by the minimization of the travel time and declining traffic congestions. The OBU contains networking, storage, and computing capabilities. In the VEC architecture, the smart vehicles are also referred to as Autonomous Vehicle (AVs), client vehicles, task vehicles, and service requestors [99].

4.1. Transforming transportation: The role of 5G networks in smart vehicle communication

The dynamic nature of vehicular networks and their diverse QoS demands have given rise to numerous challenges. These challenges are addressed through the integration of real-time information sharing applications and infrastructures, providing vehicles with access to relevant information about their immediate environment. This leads to the development of road safety and traffic efficiency applications.

Vehicular communication can be broadly accomplished through two

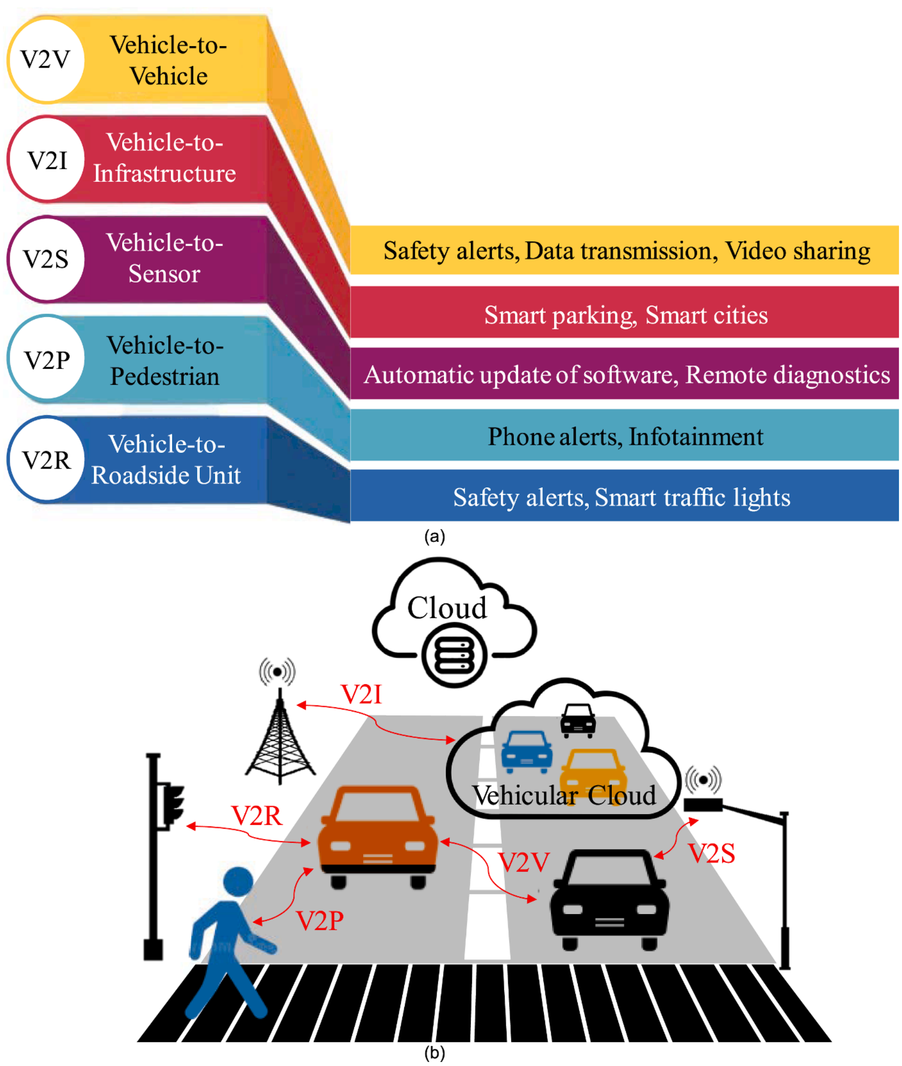

Fig. 6. (a) A taxonomy for IoV communication, (b) Vehicle-to-Everything (V2X) communication.

primary modes: V2V and V2I. These communication approaches enable various entities such as vehicles, pedestrians, and roadside units to gather data pertaining to their surrounding environment. This data is obtained by receiving information from other vehicles or sensors, fostering the development of intelligent services that focus on cooperative collision warning and autonomous driving [100]. By facilitating effective communication and data exchange, V2V and V2I communication contribute to the advancement of technologies aimed at enhancing road safety and enabling autonomous driving systems.

According to Fig. 6(a), the concept of the IoV is derived from five distinct types of network communication. These communication types serve as the foundation for enabling connectivity and data exchange within the IoV ecosystem. On the other hand, Fig. 6(b) provides a visual representation of how smart vehicles establish communication among themselves. It illustrates the comprehensive network of communication between these vehicles, highlighting the interconnectedness and exchange of information facilitated by smart vehicle technologies.

Previously discussed, communication in vehicular settings can be categorized into V2V, V2I, and V2X, encompassing interactions with both infrastructure and other vehicles. When utilizing 5G networks, the subsequent advantages should be considered [101], contrasting with IEEE 802.11p:

- Millimeter-Wave (mmWave): The utilization of mmWave technology ensures high throughput and bandwidth, which is crucial for enabling fast and efficient communication between vehicles and various entities within a dynamic and ever-changing topology scenario.
- Non-Orthogonal Multiple Access (NOMA): Through techniques such as power multiplexing or encoding, multiple users can effectively share time or frequency resources. In the context of V2X communication, NOMA holds the potential to cancel interference, thereby enhancing the capability of V2X systems to mitigate signal interference.
- Multiple Radio Access Technology (Multi-RATS): The deployment of 5G networks can bring significant benefits to V2I or V2N communications. In this context, V2N refers to vehicles communicating directly with servers or the cloud using cellular infrastructure. These benefits can be achieved through increased network capacity and throughput, which allows for faster and more efficient data transfer between vehicles and the network. Additionally, 5G networks can enhance performance in certain remote driving scenarios by providing increased redundancy, ensuring a more reliable and robust connection between vehicles and the network.
- Antenna Design: By leveraging Multiple Input Multiple Output (MIMO) technology and other related techniques, the overall capacity of the system can be increased, enabling support for a greater number of V2X activities. MIMO utilizes multiple antennas for both transmitting and receiving data, which allows for the simultaneous transmission of multiple data streams. This technique effectively boosts the capacity of the communication system, enabling it to handle a higher volume of V2X activities, such as V2V, V2I, and V2N communications. The use of MIMO, along with other advanced techniques, contributes to a more efficient and robust V2X ecosystem.
- In-Band Full-Duplex (FD): The implementation of In-Band FD technology allows for the doubling of throughput by utilizing the same frequency band for both receiving and transmitting data. In traditional communication systems, separate frequency bands are used for transmitting and receiving, resulting in potential limitations in overall throughput. However, with In-Band FD, the ability to transmit and receive data simultaneously on the same frequency band eliminates these limitations and effectively doubles the throughput capacity. This technology enhances the efficiency of communication systems, including V2X networks, by maximizing the utilization of available spectrum resources.
- Mobile Edge Computing (MEC): MEC has the potential to facilitate real-time situational awareness by enabling various capabilities, such as creating high-definition local maps and performing real-time analysis of data exchanged from multiple sources. With MEC, computational and storage resources are located at the edge of the network, closer to the end-users and devices. This proximity allows for faster processing and reduced latency in data analysis.

In the context of V2X communication, MEC can enhance the capabilities of vehicles and infrastructure to perform tasks such as generating detailed maps with high precision, taking into account real-time data from multiple sources. By leveraging the computational power at the network edge, MEC enables quick and efficient analysis of the exchanged data, facilitating timely decision-making and improving situational awareness for various applications, including ITS and autonomous driving [99].

4.2. Advantages for smart vehicles

Safeguarding traffic, enhancement of travel efficiency, and declining the emission of pollutants are among the main goals of studies on VANET technology. Regarding the commercialization issues, even the developed countries have only employed the most basic VANET technologies in the last two decades. The ineffective practical application of VANET technology can be assigned to the following reasons:

- Due to the failure to cooperate with other networks, the vehicles in the ad hoc network will lose network services upon disconnection. Thus, VANET cannot ensure continuous and stable communication.
- The incompatible network architecture of VANET has prevented its communication with several new communication devices.
- Many intelligent applications cannot be implemented due to the insufficient computing ability and storage space of VANET in addition to its lack of CC capability.
- The low precision of the application services since VANET only calculates and processes localized traffic data.

Therefore, more reliable, and market-oriented vehicle communication technologies are highly required. The deficiencies of VANET can be resolved by IoV technology, drawing promising prospects in the development of smart transportation systems. The advantages of IoV will be addressed from multiple perspectives:

i) Thanks to its heterogeneous network architecture, IoV promotes cooperation between the communication networks of vehicles and others.
ii) The majority of daily communication devices are compatible with IoV.

Efficient cooperation of various types of networks and the advent of multiple communication models (V2S, V2V, V2P, V2R, V2I) have contributed to sharing BD in addition to offering reliable communication services and expanding the scope of automotive communication applications. This is one of the most prominent benefits of IoV. In particular, V2S offers onboard sensor communication via Ethernet and Wi-Fi. V2V and V2R show the possibility of vehicles and RSU communication through WAVE. V2P indicates the communication between a vehicle and a personal handheld terminal device using Apple's CarPlay, OAA Android system, or NFC. V2I enables communication between vehicles and infrastructure through Wi-Fi or LTE/4G/5G/B5G/.

iii) Thanks to the improvement of data processing and the development of CC and AI technologies, vehicles can autonomously access the most-efficient performing networks to guarantee stable network connectivity.

4.3. Challenges in the path of development of smart vehicles

The IoV is aimed at adapting to various customers, heterogeneous networks, and vehicles and supply continuous, convenient, manageable, maintainable, and safe connections. Moreover, IoV requires several issues due to their differences compared to other networks. These requirements pose various challenges in the development of IoV [102]. The great dynamics and mobility of the vehicles and frequent topology alterations led to poor network connection and stability and network and connection failures, hence the loss of packets. Consequently, a major difficulty is ensuring reliable connectivity and coverage. IoV applications have latency restrictions in terms of delays, but they don't call for large data rates. For instance, if someone breaks on the highway, in-time notifications must be sent to the other vehicles to avoid accidents. In these situations, the minimum delay, not the average delay, determines the outcome.

In terms of service sustainability, delivering sustainable service in the IoV environment and the network mechanism's user-friendly design is a significant problem. The establishment of continuous services by heterogeneous networks in a real-time environment is a critical obstacle as these services possess special network bandwidth, limited-service platforms, various wireless access, and complex urban structures [103,104].

The most difficult problem for every IoV application is to safeguard the user's security and privacy. For reliable information to be obtained from its originating point to its endpoint, the relevant user must be authorized. IoV continues to face a serious problem with protecting private data breaches, leaving it open to cyberattacks.

5. Offloading in 5G-enabled VEC system

5.1. Motivation for offloading

Considering the growing number of vehicles and the rapid expansion of the IoV, vehicles are now a remarkable part of internet-connected things. The IoV paradigm provides smart automobiles with intelligent vehicle control, traffic management, and interactive applications which may require considerable computation resources at short delay [105]. Nonetheless, automobile terminals have limited computational capability. Thus, applications demanding large computation resources may be a great challenge for the vehicular terminals with limited resources.

In this context, the novel paradigm of cloud-based vehicular networking has been introduced to handle the huge computation demands of automobile terminals and improve service performance. Through the integration of communication and computing technologies, cloud-enabled networks enable applications to run locally on the vehicle terminals or be offloaded to a remote cloud.

MCC can remarkably improve resource utilization and computation performance. Nevertheless, locating the cloud servers far away from the mobile vehicles might seriously deteriorate the offloading efficiency due to the capacity limitations and delay fluctuations of the transmission on the backhaul and backbone networks. MEC is thus developed to derive cloud service to the edge of the radio access network and offer cloud-based computation offloading near the mobile vehicular network.

The computational tasks of the MEC networks may have various resource requirements (the computation resources for task execution and task transmission). As MEC servers function at the edge of the radio access network and transmit the tasks through connected RSUs, their service scope might be limited by the radio coverage of the RSUs. Regarding the high mobility of vehicles, they may pass several RSUs and MEC servers during the task-offloading process, therefore, they could offload their task to any MEC servers in access. The choice of the MEC servers can influence the offloading efficiency.

Moreover, vehicles can employ several methods (V2I and V2V modes) to access RSUs in connection to MEC servers. The dynamic topological variations due to the mobility of vehicles can add to the complexity of offloading transmission. An optimal task-offloading scheme with MEC server selection and communication management is required in the MEC cloud-based vehicular networks to enhance task accomplishment efficiency.

CC infrastructures have been utilized to handle resource-demanding applications. Cloud resources are, however, located far from the users which may lead to bandwidth problems, failure in supporting delay-vulnerable applications, and security and privacy issues [106]. Therefore, the resources should be brought closer to the network edge for complete support of dynamic scalability, network processing efficiency, and better design of computing paradigms [107]. Accordingly, Multi Access Edge Computing (MAEC) can decrement the latency and save energy by offloading computation on the edge servers [108]. The conventional MEC-based offloading approach may fail in the vehicular environment due to the high mobility and short validity period of fast mobility [109].

Considering the dynamic and latency-sensitive nature of vehicular networks, EC-based offloading strategies emerge as the most suitable. Unlike traditional cloud-centric approaches, EC allows for data processing and analysis closer to the point of data generation, minimizing latency and bandwidth consumption [110]. In vehicular environments, where real-time decision-making is crucial for safety and efficiency, EC offers low-latency processing and reduced dependency on centralized cloud resources [58]. Furthermore, EC enables localized processing, which is essential for addressing connectivity challenges and ensuring continuity of services in vehicular networks [111]. Thus, the integration of EC with the IoV presents an effective offloading strategy, enhancing resource management and computational capabilities while mitigating challenges associated with network connectivity and latency-sensitive applications.

Current attempts are now focused on merging the MEC technology into a vehicular network. In particular, VEC is an MEC technology for vehicular networks. VEC is specifically beneficial for computation-demanding and time-constrained tasks [97]. Computational latency and energy consumption of vehicular applications can be remarkably decremented at a reduced chance of network congestion by offloading complex computational tasks over VEC servers. Task offloading to edge servers may be sometimes infeasible due to high energy consumption and long process times [112]. The challenge involves making the best offloading decision considering overall computational and communication costs. Moreover, vehicles may encounter unprecedented constraints such as inadequate computational capacity and high energy consumption [113]. These constraints may lead to the following scenarios:

- Meeting the real-time stringent time and energy demands of the vehicles regarding their limited computational and energy resources.
- To guarantee mileage durability of the vehicles especially autonomous ones with high energy demand for computation-intensive applications.
- Efficient management, transmission, and storage of massive data generated by autonomous vehicles.
- Coping with high hardware costs as the vehicular computing capabilities cannot bear the growing computing demands.

The above-mentioned challenges can be well addressed by task offloading approaches.

5.2. Task offloading techniques

The offloading strategy aims to decide whether the vehicle should engage in offloading and the extent of the offloading. Typically, there are three outcome categories associated with offloading policies in VEC: full, partial, and binary task offloading. The determination of the task vehicle's outcomes is influenced by both the vehicle's energy delay and the computing task's time delay. The objectives of the offloading strategy can be encapsulated in several dimensions, including minimizing delay, decreasing energy consumption, achieving a balance between delay and energy, and optimizing overall system performance.

1. Full task offloading:

Full task offloading involves transferring the entire computational task from a local device (e.g., a mobile device or vehicle) to a remote server or cloud. This approach is suitable for tasks with high computational demands and minimal dependency on local resources. It can help offload the entire burden of computation, freeing up local resources for other applications [114].

2. Partial task offloading:

Partial task offloading focuses on dividing a computational task into segments, where some segments are offloaded to remote servers or the cloud, while others are processed locally. This approach is useful when certain parts of a task can be efficiently processed locally, and only resource-intensive segments need to be offloaded [115].

3. Binary task offloading:

Binary task offloading involves making a decision on whether to offload the entire task or keep it entirely local. This decision is typically based on specific conditions, such as the availability of network resources, the computational capability of the local device, and the urgency of task execution. It provides a binary choice between offloading or keeping the task on the device [13].

5.3. Computing offloading

Computation offloading refers to the offload of computationally expensive or delay-vulnerable tasks to edge devices or nearby edge servers to efficiently ensure high user service quality [116]. The computation offloading is mainly aimed at reducing the response delay and improving the service quality. Furthermore, the overall performance of the system can be enhanced by transferring the computation tasks to the edge server or the cloud data centers in cases with insufficient processing capability of the edge node. Diverse issues (e.g. performance maximization at minimal energy consumption) should be included while making the computation offloading decisions. The following queries must be addressed before computation offloading:

- Is it possible to offload the task? The task scheduler determines the possibility of offloading the task, i.e., what is offloaded; partially or totally?
- When to offload? The task scheduler determines the time of offloading considering various constraints.
- Offloading place; where is the best location to offload the workload execution, considering available resources distribution.
- What is the offloading procedure? The basic goals of task offloading, single performance maximization, or joint optimization and trade-off between many goals are all relevant to this query. For instance, the architectural, performance, and power supply heterogeneity of massive edge devices can lead to non-uniform distribution of EE among devices. Additionally, dynamic variations of network bandwidth and latency among cloud data centers and edge equipment can alter energy consumption of the data transmission. Thus, various computation offloading policies lead to differences in the power consumption. The overall computation delay, data transmission, and performance metrics must be ideally balanced in a good computation offloading policy. Current trends in computation offloading, as well as challenges and future research directions, will be discussed in the continue.

The implementation of an offloading system, resource allocation, and offloading policy are all included in ComOf, as shown in Fig. 7. The procedure for unloading is as follows:

- A service request: Within its communication range, the task vehicle transmits a service request to the infrastructure (e. g. base station, RSU).
- Upload tasks: Through V2E connection, the task vehicle transmits the computing tasks to edge servers.
- Execute tasks: The edge server decides to offload duties to another location in accordance with resource allocation strategy to complete computing tasks.
- Deliver outcome: The computational output is returned to the vehicles via the edge server.

The proximity of edge nodes to the vehicles in VEC offers a compelling reason to incorporate EC into the architecture, primarily to tackle the issue of latency. In high- and low-density networks, one approach to mitigate this problem is through data computation offloading. Typically, significant delays are experienced when vehicles transmit data to the central cloud [117].

As depicted in Fig. 8, both RSUs and vehicles, whether in motion or parked, can serve as edge nodes and contribute computing resources within the VEC framework. Client nodes, including vehicles that require additional computing power, onboard User Equipment (UEs), and pedestrians, generate computation tasks with varying workloads and delay requirements. These tasks are offloaded from the client nodes to the VEC for further processing. Noteworthy, it is important to note that each vehicle can assume the role of an edge node or a client node, referred to as an *edge vehicle* or a *client vehicle*, respectively. The role of each vehicle may change over time, meaning that it may have excess computing resources to share with the network as an edge vehicle or require support from other nodes as a *client vehicle*.

1) What to offload: Offloaded workload's description

The objectives of EC to reduce the service latency and network bandwidth cannot be accomplished unless the workload in cloud data centers can be offloaded to edge devices and servers. The original workloads performed at cloud data centers must be divided in edge-cloud coexisting environments, and part of them should be operated on edge devices and servers. The distribution of local computing resources across a large number of heterogeneous devices, such as IoT devices, is non-homogeneous. This non-homogeneity often leads to insufficient computing resources to effectively run complex programs. Lower latency and improved system performance can be achieved with careful offloaded workload selection.

There are several ways to increase the Quality of the User's Experience (QoE), including caching content data close to the end users, outsourcing data processing to proxy servers, caching dynamic page fragments, and conducting page composition after the user has accessed the page.

Based on the Lyapunov optimization, Z. Ning et al. [67] developed the online multi-decision making algorithm known as OMEN, which can function independently of future system information. Theoretically, it has been demonstrated that OMEN performs optimally within a defined deviance. The function of different RSUs was then analytically described to create the best peer offloading strategy using the Lagrange multipliers method. J. Xu et al. [118] proposed an algorithm named online service caching for mobile edge computing (OREO) for online stochastic service caching with no need for future information. OREO was developed on the basis of the Lyapunov optimization and managed to achieve close-to-optimal performance in comparison with the optimal algorithm with full future information, while avoiding the violation of energy constraints.

Offline prefetching, also known as traditional ComOf, entails moving user-entered data from the edge device to the cloud data center or edge

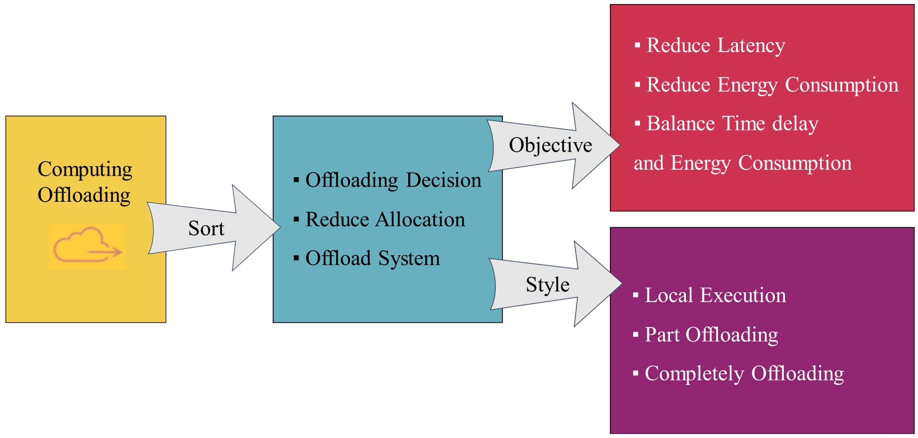

Fig. 7. Integrated offloading system with resource allocation and offloading policy.

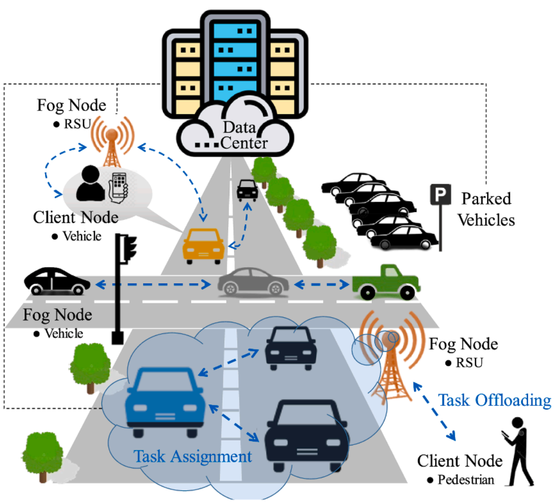

Fig. 8. Task offloading in the VEC.

server before processing, which can generate a lot of network communication traffic.

D. Han et al. [119] proposed a unified framework to minimize the overall outage probability in different mobile computation offloading scenarios. Specifically, the outage bottleneck is determined through asymptotic analysis, with no need for precise outage probabilities for both transmissions and computations. Resource pairing, matching, and allocation policies are explored to cope with the outage bottleneck. Theoretical analysis, as well as numerical findings, indicated the dependence of the outage bottleneck not only on the accessibility of spectrum and computation resources but also the probability distributions of computational complexity of the tasks.

2) When to offload: The exact timing

To reduce latency for services that require processing, computation offloading makes use of the computing, storage, networking, and energy capabilities of edge devices. Nonetheless, regarding the continuous changes in network conditions, decisions on workload offloading should specify when to offload the workload. To put it another way, the task scheduler must precisely plan the timing of the offload while taking into account all circumstances and system status. For instance, during times of network congestion, data caching can significantly improve system performance while easing the transfer of huge amounts of data to cloud data centers over adequate lines.

The issue of offloaded workload selection was addressed in the previous section including data caching, data storage and computation and analysis. Here, the offload timing is discussed. The question of when to offload can be inferred from the precise temporal slots at which workload offloading produces the optimum performance at minimum costs or overheads. Upon deciding on the computation offloading, the data and task are partitioned. Proper and accurate scheduling of task offloading can effectively improve the system performance while minimizing resource utilization because of the dynamics of network connection and availability of edge devices. The performance of the system might also be impacted by the partitioned workload's execution sequence. Therefore, improved offloading decisions can result from system monitoring and workload evaluations based on task arrival rates and deadlines.

3) Where to offload: Offloaded workload's scheduling

During the workload offloading process, partitioned tasks are scheduled to specific edge servers and devices based on various factors such as performance, network bandwidth, energy requirements, and data privacy protection strategies. The selection of targeted edge devices and servers takes into consideration these factors to optimize the overall system performance. For example, energy-intensive tasks are offloaded to cloud servers to conserve energy, as these servers are typically equipped with higher computational capabilities and can handle such tasks efficiently. On the other hand, data-intensive processes are offloaded to edge servers, which are closer to the edge devices, to reduce latency and minimize network traffic. This approach improves the overall responsiveness of the system by ensuring that data-intensive tasks are processed closer to the data source. Additionally, data privacy protection strategies are considered to ensure the security and confidentiality of sensitive data during the offloading process.

When scheduling offloaded tasks, it is crucial to consider the overall system status, which includes the network status, task requirements, and device information. This comprehensive approach ensures that tasks are assigned to the most suitable computing resources. For instance, when there is sufficient network bandwidth available, cloud servers can be utilized to perform workloads efficiently. Cloud servers often possess high computational power and can handle resource-intensive tasks effectively. However, if the network bandwidth is limited or the latency requirements are stringent, edge servers or local devices are more appropriate options. Edge servers, being closer to the edge devices, can provide low-latency processing and reduce the dependence on the network infrastructure. By considering the system status and the specific needs of each task, the scheduling process optimizes the allocation of tasks to the most suitable computing resources, ensuring efficient execution and meeting the desired performance objectives.

The best offloading decisions are made upon determining the offloading place to enhance the system performance for achieving maximal efficiency. Such an offloading decision, however, requires periodic monitoring of several parameters which could be computationally expensive, leading to additional overheads upon running on a mobile device. To overcome this problem, W. Junior et al. [120] developed a Context-Sensitive Offloading System (CSOS) which utilized the main machine-learning reasoning techniques and robust profiling system for offloading decision-making with excellent precision. In their study, the researchers evaluated different classification algorithms for their database and found that the JRIP and J48 classifiers achieved an accuracy rate of 95 %. They further extended their investigation to include controlled and real scenarios, where the context information varied between different experiments. In these varying conditions, the CSOS system demonstrated the ability to make accurate decisions while also improving overall performance and EE. The results indicated that CSOS effectively adapted to changing contexts and provided reliable decision-making capabilities while achieving performance gains and energy efficiency objectives.

4) A tradeoff between energy and QoS in data computation and communication

With the ever-increasing use of applications on smart mobile devices, the user's QoS can serve as a prominent indicator of the success of the applications and devices. These smart mobile devices often possess limited computing and storage resources, as well as battery capacity, hindering the satisfaction of the growing demands of mobile users. Better quality services can be achieved through scheduling resources based on the user requirements and Services Level Agreements (SLAs). In this way, delay-sensitive applications are prioritized while computation-demanding applications are provided with sufficient computing resources. In this regard, the QoE can represent the user's subjective perception on the QoS and device performance. Computation offloading to the edge servers and then resending the results to the mobile devices can remarkably modulate the resource demand of the smart mobile devices. During computation offloading requires implementation of QoS and QoE requirements, formulating a reasonable task offloading sequence, and determination of the offloading timing for each task [121,122]. Computing offload to mobile devices can save energy and enhance the processing power, however, communication between the involved components (e.g. mobile devices, edge nodes, and cloud servers) leads to execution delays, affecting the performance of the application. This highlights the significance of establishing a balance between computing and communication.

To enhance the efficiency of the edge node system in terms of data storage and real-time data access, one approach is to deploy high-density, low-power, low-latency, and high-write nonvolatile storage media, such as Nonvolatile Memory (NVMe), at the edge device. This enables efficient and uninterrupted storage of data while ensuring continuous and real-time access to it. Additionally, power profiling and accounting support provided by the system call and runtime library can be utilized to reduce power consumption during runtime. By optimizing the power usage of the code, the programmability and EC of the edge node system can be increased, contributing to improved overall performance and sustainability.

5.4. Offloading process

The offloading process determines when and how much to offload from the vehicle. Offloading policy VEC, local execution, total offloading, and partial offloading policies typically produce three different types of outputs. The task vehicle's outcomes are influenced by the vehicle's energy delay and the computing task's time delay.

Offloading strategy's main objectives can be perfectly described as minimizing delay, decreasing energy consumption, and balancing delay and energy consumption. In Fig. 9, user tasks, depicted in the illustration, have the option to be performed both locally within the VEC and transferred to the network edge, which is enhanced by cloud capabilities in terms of computation and storage resources. This process significantly reduces delay and eases the burden on the network infrastructure.

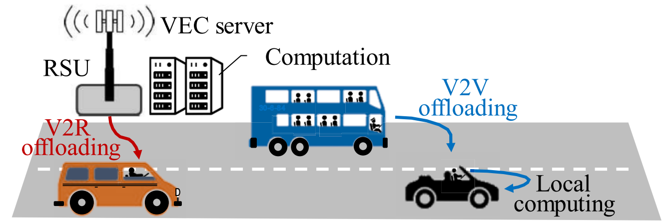

Fig. 9. Example of task offloading.

- Local Operation: The vehicle itself completes the computing task.
- Approved offloading: The computing task is delegated to RSUs, who processes it.
- Restricted offloading: The RSUs server processes the remaining portions of the computing task while some of it is handled locally.

As illustrated in Fig. 10, the delay, energy cost, and delivery of task vehicles under various offloading policies are stated.

The offloading process must take into account the computing latency since it may compromise the QoE. Additionally, the issue of energy consumption must also be considered. The battery of the mobile device terminals will run out if the device uses too much energy.

6. Classification of offloading

The widely-scattered computing resources of a Vehicles Edge Network (VEN) offer diverse offloading routes. Various communication techniques have been jointly utilized to support data transmission between client and edge nodes; among which IEEE 802.11p-based Dedicated Short-Range Communications (DSRC) and LTE-V can be mentioned which supports vehicle-to-everything (V2X) communications such as V2V, V2I, and V2P communications. In addition to vehicles, pedestrians can also access RSUs through 3G or 4G LTE networks. This enables pedestrians to connect to the vehicular network and access the services provided by the RSUs. Furthermore, onboard UEs, such as smartphones or other mobile devices, can offload their tasks to the vehicles they are traveling in using Bluetooth technology. This allows UEs to leverage the computational resources of the vehicles for offloading tasks, enabling efficient processing, and enhancing the capabilities of the UEs. By supporting connectivity options like 3G, 4G LTE, and Bluetooth, the vehicular network extends its reach to include both pedestrians and onboard UEs, facilitating seamless communication and task offloading across different entities in the network. The V2V mode of communication is depicted in Fig. 11. Applications like Road-Accident and Street Parking leverage the cooperation between vehicles and roadside units to extend the communication range of the vehicular network when direct V2V communication is not feasible. In scenarios where vehicles are out of range for V2V communication, these applications rely on other vehicles to act as intermediaries. These intermediary vehicles receive information from the source vehicle and then forward it to RSU within its communication range. This collaborative approach enables the sharing of critical information, such as road accidents or available parking spaces, by leveraging the relay capabilities of vehicles and the coverage provided by RSUs. By utilizing these intermediary vehicles, the communication range of the vehicular network is extended, facilitating effective information exchange, and enhancing overall system efficiency.

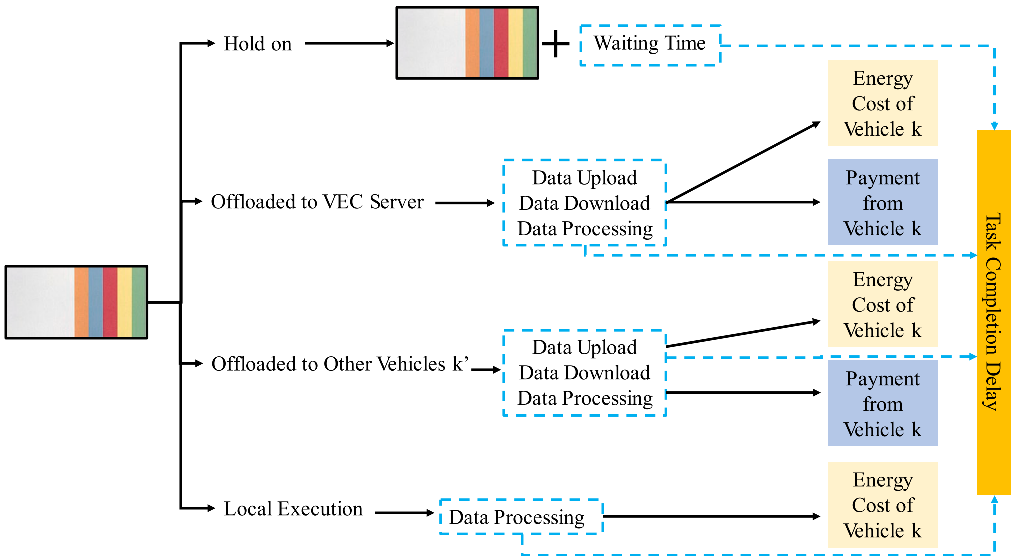

Fig. 10. Offloading processes: delay, energy cost, and delivery.

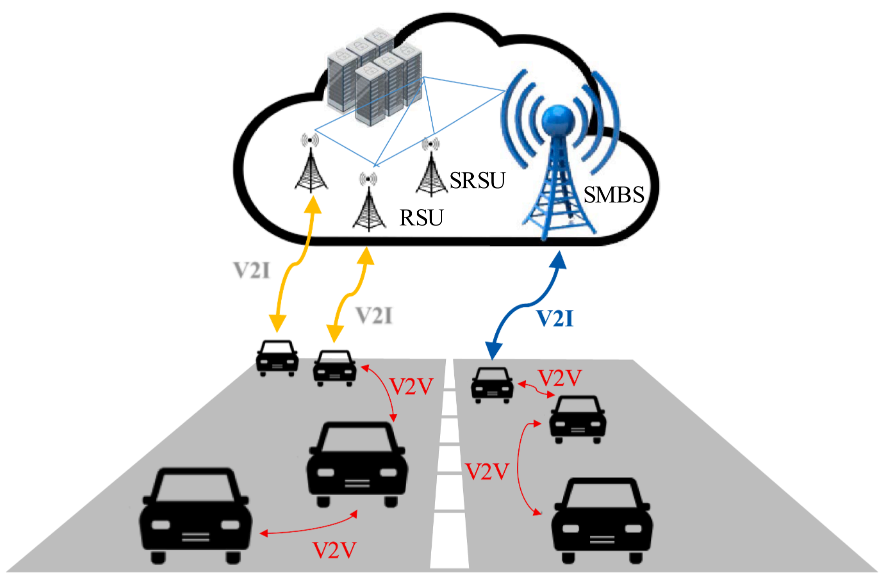

Fig. 11. V2V and V2I communications.

6.1. Data offloading through Vehicles-to-Vehicles (V2V) communication

Vehicles can directly offload their tasks (and also the tasks offloaded by their passengers) to adjacent edge vehicles. Each client vehicle finds accessible edge vehicles in its communication range. Note that the direction and speed of the movement should be considered (obtained by V2X communication protocols) to maintain a relatively long contact time. Several edge vehicles may be simultaneously available. Each client independently decides to select which edge vehicle, as it is difficult to obtain global information, moreover, there might be no centralized entity for making such decisions.

The computing units integrated within vehicles have the capability to support applications that require low latency and extensive computation. These vehicles can function as edge servers, offloading computational tasks to reduce the burden on infrastructure nodes. When vehicles are in close proximity, they can establish V2V connections to offload their tasks, either completely or partially. This approach, referred to as V2V-based computation offloading, is depicted in Fig. 12-A.

In a study [123], two vehicles communicated with each other by a cellular network. The EC server employed the network state routing to discover the V2V offloading path with the longest communication life. Upon disconnection of the corresponding V2V path, the two vehicles switched back to the cellular network for communication. IDM IM model was employed in a study [124] to model the vehicle movement. Vehicle tasks were divided and offloaded to adjacent vehicles. The maximum and minimum fair algorithm was utilized to determine the extent of offloading in a special scenario while the Particle Swarm Optimization (PSO) algorithm was employed to obtain the offloading scheme within a general scenario. This approach minimized the delay. However, the neighboring vehicles in this paper were not within the one-hop communication range of the VT. A task offloading scheme was

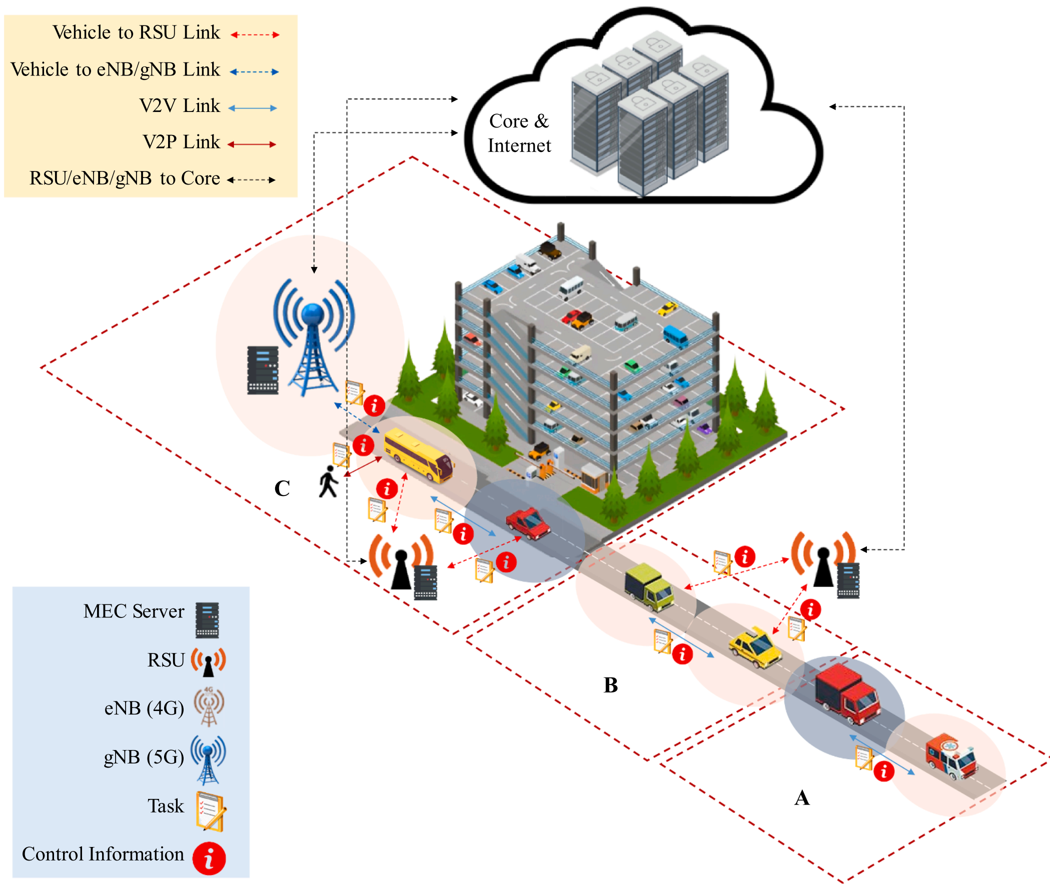

Fig. 12. Computation offloading scenarios in 4G and 5G RAT systems (i.e., RSU, eNB, and gNB): Exploring various communication modes defined as scenarios A: for Vehicle-to-Vehicle (V2V) communication, B: for Vehicle-to- Infrastructure (V2I) communication, and C for Vehicles-to- X (V2X) / mixed offloading.

also developed [125] by utilizing multi-hop vehicle computation resources in VEC on the basis of vehicle mobility analysis. Besides the vehicles in one hop from the task vehicle, certain multi-hop vehicles meeting the link connectivity and computation capacity requirements were also utilized to complete the offloaded tasks. An optimization problem was also considered for the task vehicle to minimize the weighted sum of execution time and computation costs. To gain insights into the computational offloading schemes in the V2V communication domain, Table. 5 highlights the distinct features and contributions of the research.

6.2. Data offloading through Vehicles-to-Infrastructure (V2I)

Within the framework of conventional offloading strategies, the vehicle can only offload their tasks within the communication range of the EC server. Such an offloading approach may fail in meeting the offload delay requirements. This approach considers the vehicles within and outside the communication range of the EC server as the destination and source nodes, respectively.

Vehicles access VEC servers through V2I communication links. Vehicles with local computing resources cannot perform computation-demanding and complex tasks on their own [126]. VEC can supply the heavy vehicular computing demand by offering computing capabilities at the edge of the network. To this end, a vehicle connects the roadside infrastructure (like RSUs or BS) via V2I for entire or partial offload of its task on the MEC server as depicted in Fig. 12-B.

An assistant algorithm was utilized in V2V [127] to predict the arrival time to the subsequent RSU. In this system, the tasks are transferred through V2V to the subsequent RSU. The results can be directly acquired upon the arrival of the vehicle to the next RSU. In comparison with the task offloading to the current RSU, it is more cost-effective to exchange the results between RSUs when the vehicles move and arrive at the next RSU and then return to the vehicle. However, it is not reasonable to consider a fixed value for the cost of each hop. It does not consider situations such as link disturbance due to V2V transmission. Wireless interference and transmission capacity limitations were not investigated for the simplicity of the unloading process. To improve service reliability, a deep learning-inspired RSU Service consolidation solution [128] based on two models was created. The RSU Migration model and the Multicast model based on linear programming are used to define the RSU coverage issue and content delivery challenges, respectively. On the basis of content correlation, an adaptive packet-error measurement system can also be used to improve service reliability rates at the edge of cooperative vehicular networks. Table. 6 provides a comprehensive summary of the latest research contributions in the category of V2I-based computation offloading.

6.3. Data offloading through Vehicles-to-X (V2X) / mixed offloading

V2X communication, encompassing various modes like V2V, V2I, V2P, and V2N, proves to be the most effective approach for offloading computations in vehicular networks. In the context of computation offloading, vehicles can engage in communication with nearby vehicles (V2V), roadside infrastructures (V2I), and pedestrians (V2P) to offload tasks completely or partially for computation. This method, referred to as V2X-based computation offloading, is depicted in Fig. 12-C.

Task offloading highly depends on the task type. The tasks demand low computations, but a huge amount of task data can be locally accomplished. The tasks with large computation demand and low data transmission requirement can be offloaded to edge servers, the remote

Table 5
Overview of V2V computation offloading schemes.

Object	Scheme	Computing Architecture	Advantages	Disadvantages	Proposed solution
Minimization of Latency	Distributed [129]	VEC	Establishing an Adequate System Model	Assumption of Equal Input/output Sizes	To achieve reduced latency, tasks are duplicated and transferred to multiple SeVs (Service Vehicles) while employing the framed LTRA (Latency-Aware Task Replication and Adaptation) algorithm, which can dynamically adapt to the time-varying topology of the VEC system.
	Centralized [130]	VEC	Optimum use of vehicle resources	There is no consideration for different service types	Developed an innovative on-policy reinforcement learning framework for computation migration that adapts to changing environments through continuous learning.
	Distributed [131]	VCC	Notable performance against expectations	large overhead signals. Not taken into account is the high-density vehicle scenario	Through the utilization of SMDP (Semi-Markov Decision Process), an offloading policy that is sensitive to the requirements of the application is implemented to derive an optimal scheme for allocating computation resources.
	Distributed [132]	VCC	Optimal use of a vehicle's resources	A lack of thought went into choosing the vehicles. Application completion time uncertainty	To support time-sensitive applications with a sequential task graph, a computation-relaying system was investigated.
	Distributed [133]	VEC	Optimum use of the resources available to vehicles	Possible major delay due to the number of offloading hops	Without the use of RSUs, the vFog framework uses the onboard processing capability of the cars to communicate with surrounding vehicles.
	Distributed [134]	VCC	Analyzing real-world mobility traces for comprehensive mobility assessment.	Sophisticated model architecture	A three-layer architecture has been devised in the IoV framework, employing Deep Reinforcement Learning (DRL) techniques. The primary objective is to minimize overall energy consumption while simultaneously adhering to specified delay constraints.
	Distributed [124]	VCC	Satisfying the demands of delay-sensitive applications through collaborative vehicular computing approaches.	Insufficient examination of the influence of vehicle mobility on offloading performance analysis.	A task offloading strategy leveraging V2V communication is developed to effectively utilize idle resources in vehicles. The problem is optimized using the Particle Swarm Optimization (PSO) algorithm.
	Distributed [135]	VEC	Adapting to dynamic environmental conditions and mobility patterns.	Absence of comprehensive analysis regarding mobility patterns and their impact	Introduced a novel deep Q network algorithm aimed at minimizing both delay-related costs and energy consumption.
Resource Allocation	Distributed [136]	VCC	A realistic mobility scenario reflecting actual movement patterns and dynamics.	Significantly extended computation time or prolonged computational processing.	The AVE (Adaptive Vehicular Edge) framework is developed to enhance the computational capabilities of vehicles. This framework efficiently manages the underutilized computing resources without relying on centralized control.
	Distributed [137]	VEC	A well-structured and organized system model that meets the desired criteria.	Lack of inclusion of mobility analysis based on real-world traces in the current approach.	A collaborative computing scheme is proposed, where a group of autonomous vehicles (AVs) is utilized to enhance the scalability and efficiency of autonomous driving. This scheme leverages Software-Defined Networking (SDN) principles to facilitate effective communication and coordination among the vehicles
	Centralized [138]	VEC	Ensuring system stability and achieving low latency through efficient resource utilization.	Inadequate performance under high mobility conditions resulting in suboptimal outcomes.	Proposed a collaborative task scheduling scheme for computation offloading in Vehicular Cloud (VC) environments using a modified genetic algorithm with low complexity.
	Distributed [139]	VEC	Limited resources in densely populated areas.	Limited resources in densely populated areas.	Developed a framework for offloading based on machine learning principles and Multi-Armed Bandit (MAB) theory. The framework enables vehicles to learn from nearby vehicles with sufficient resources, enhancing offloading performance.
	Centralized [140]	VEC	The payment procedure for the seller, specifically the vehicle fog node, is established.	Prioritizing offloading to the seller instead of the local vehicle as the initial option.	A centralized reverse auction mechanism based on VCG (Vickrey-Clarke-Groves) is devised to ensure individual rationality and truthfulness while considering economic aspects.
	N/A [141]	VEC	Achieving the optimal decision for task offloading.	The self-organized and ad-hoc nature of the system results in challenges related to security and resource management, which are not adequately addressed.	A V2V computation offloading approach is introduced in the context of 5G cloud-enabled IoV, considering scenarios with both complete and incomplete information.

(*continued on next page*)

Table 5 (*continued*)

Object	Scheme	Computing Architecture	Advantages	Disadvantages	Proposed solution
	Distributed [142]	VEC	Rapid convergence rate	Limited resource availability in densely populated regions	Proposed an adaptive learning-based offloading algorithm utilizing the principles of Multi-Armed Bandit (MAB) theory. To account for dynamic environments, the algorithm incorporates input-awareness and occurrence-awareness. This enables efficient offloading of a vehicle's tasks to nearby vehicles with sufficient resources.
	Centralized [143]	VEC	Efficient utilization of available computing resources.	Applicable to modest and small-scale scenarios	A task offloading framework based on DRL is proposed to optimize the cumulative reward of task vehicles. The framework takes into account various factors such as the state of V2V links, service vehicles' motivation, costs associated with tasks and service vehicles, and the availability of resources in service vehicles, efficiently maximizing the overall reward.
	Distributed [144]	VCC	Improved user QoE and QoS	Unreasonable assumptions made by randomly selecting arrival rates	Presented a Reinforcement Learning (RL) algorithm for the self-adaptive resource allocation in the IoV environment. The algorithm takes into account both Semi-Markov Decision Process (SMDP) and Markov Decision Process (MDP) models to optimize resource allocation.
	Distributed [145]	VEC	Formulating an approach that meets the required criteria and conditions.	The introduced DL model does not account for the inference error that it may introduce.	A joint optimization scheme is formulated, taking into account road metrics and resource management. The scheme utilizes a sleeping multi-armed bandit tree-based algorithm to efficiently manage resources.

clouds, or other idle vehicles [197]. Thus, a combination of several offloading schemes can be simultaneously taken into account for the vehicles handling multiple types of tasks.

The pure V2V offloading has been rarely explored. The current V2V offload is mainly concentrated on routing designs which involve routing the MEC-uncovered vehicle tasks to the closest MEC. This scheme is not, however, suitable in scenarios with poor MEC coverage. Additionally, the current method restricts the range of vehicles supplying computational offloading service to two hops in a distributed V2V offloading scenario [6,124]. A joint frequency scheduling and power control scheme was also developed in [198] to improve the connectivity of multi-hop V2I/V2V networks. V2I and V2V links were associated with tuple-links. Then, an NP-hard problem was formulated where the frequency scheduler and power controller were collectively designed for the tuple-links.

Adiththan and colleagues [199] proposed an adaptive data offloading technology for CC control calculations to achieve a ComOf technology for vehicle safety and stability requirements in the presence of unreliable communication networks. This method took current network conditions and control application requirements into account to determine the feasibility of remote computing and storage resources. In the meantime, a cloud-based path was described utilizing crowdsourced data for path planning. Accordingly, the authors developed an adaptive offloading controller architecture capable of determining the offloading time of control calculations on the cloud such that additional data and computing resources could be utilized for the implementation of CC. The feasibility of the developed method was confirmed by a cloud-based path controller using Matlab.

Wang and colleagues [187] developed a ComOf scheme that combined FC with a decentralized traffic management system for real-time traffic management in a fog-based IoV system to minimize the average response time. To this end, they initially developed a distributed urban traffic management system where vehicles in proximity of RSUs can be utilized as fog nodes. Based on the queuing theory, they subsequently modeled the parking and moving vehicle-based fog nodes and came to the conclusion that the mobile vehicular fog nodes could be modeled as M /M /1 queues. At last, an approximate method was proposed to optimize the offloading problem by decomposing the optimization problem into two sub-problems and scheduling the traffic flows in different fog nodes. The performance of the proposed method was explored by simulations considering the real-world tax trajectory dataset. The results indicated the superior performance of the proposed method over conventional methods.

Table. 7 provides a comprehensive overview of the state-of-the-art research contributions in the V2X category, facilitating readers' understanding and providing a clear snapshot of the advancements in the field.

7. Task offloading in dynamic Edge-IoV networks

In the landscape of Edge-IoV networks, the dynamic nature of vehicular movement poses unique challenges to the effective implementation of task offloading strategies. This section aims to delve into the complexities and considerations associated with task offloading in dynamic scenarios, where vehicles are in constant motion. Understanding how task offloading adapts to the dynamic environment of Edge-IoV networks is crucial for optimizing computational efficiency and ensuring seamless operation.

The continuous movement of vehicles introduces unprecedented challenges to the task offloading process. Factors such as varying signal strength, intermittent connectivity, and frequent handovers between different network elements significantly impact the decision-making process of when, where, and how to offload tasks. In Fig. 13, we illustrate several key challenges in VEC along with their interconnections. These challenges have been acknowledged in the literature, emphasizing the need for tailored solutions in the context of dynamic IoV environments [200,201].

To confront these challenges, innovative strategies have been proposed to enhance the adaptability of task offloading mechanisms in the presence of continuous vehicle movement. Predictive modeling for vehicle trajectories, adaptive offloading triggers based on real-time vehicle data, and dynamic resource allocation methodologies have emerged as key strategies [184,202]. These adaptive strategies aim to optimize task offloading decisions in real-time, accounting for the

Table 6
Overview of V2I computation of offloading schemes.

Object	Scheme	Computing Architecture	Advantages	Disadvantages	Proposed solution
Load Balancing, Resource Allocation, and Server Selection	Centralized [78]	MEC	A low-complexity system model that is deemed acceptable and manageable	A mobility model that incorporates stochastic elements, accounting for random variations and uncertainties in the movement patterns of entities.	A low-complexity algorithm is introduced to enhance server selection, optimizing the offloading ratio and computation resource allocation.
	N/A [146]	MEC	A well-designed and appropriate system model that meets the required specifications and criteria.	suboptimal conditions in vehicular environments.	Introduced a stochastic optimization model utilizing dynamic programming, along with a data transmission scheduling scheme. This approach aims to maximize the lower bound of performance while accounting for the inherent randomness in V2I communications.
	Centralized [51]	MEC	Improve the probability of task success.	inadequate concluding remarks.	An introduced mobility-aware greedy algorithm is utilized to determine the allocation of edge cloud resources to individual vehicles. This algorithm aims to achieve near-optimal performance while enhancing the probability of successful task execution.
	Distributed [147]	MEC	Enhances the overall utility of the system.	A general vehicular mobility model that captures the movement patterns and dynamics of vehicles in a realistic manner.	Developed the DCORA (Distributed Cooperative Offloading with Reduced Complexity Algorithm) algorithm, which effectively reduces the overall system complexity.
	Distributed [148]	MEC	The approach proves to be effective in reducing latency and improving network connections.	The proposed approach results in an increase in computational complexity and poses challenges in achieving convergence.	A scheme based on Multi-Agent Deep Deterministic Policy Gradient (MADDPG) is proposed to manage the spectrum, computing, and caching resources allocated to the MEC −mounted MeNB (Mobile Edge NodeB) and Unmanned Aerial Vehicles (UAVs).
	Distributed [149]	MEC	The framework enables distributed knowledge sharing and facilitates knowledge reuse between agents, thereby accelerating the learning process.	The computational complexity exhibits an exponential increase as the number of agents in the system grows.	A multi-agent deep Q-learning approach is proposed and applied to maximize the utilization of communication and computation resources.
Minimization of Energy	N/A [150]	MEC	The proposed scheme exhibits adaptability to variations in vehicles' speeds and changes in the wireless transmission environment.	The current approach demonstrates inadequate performance when faced with complex vehicle mobility scenarios.	A dynamic offloading scheduling scheme is introduced specifically for vehicular networks, taking into account the limited resources and mobility constraints of the vehicles.
	Centralized [151]	MEC	Efficient utilization of energy through offloading and power control strategies.	The current approach lacks an analysis of mobility patterns or considerations of vehicle movement dynamics.	An energy-efficient resource allocation algorithm based on Alternating Direction Method of Multipliers (ADMM) is developed for in-vehicle User Equipment (UEs) with limited battery capacity.
	N/A [152]	MEC	A system model that meets the required standards and criteria, considered suitable for the given context.	Absence of a dedicated mobility model.	An approach is introduced for dynamically making task offloading decisions, allowing for flexible subdivision of tasks. This aims to minimize energy consumption and reduce packet drop rates.
	Centralized [153]	MEC	Expediting Convergence	Not validated in a network environment.	An adaptive offloading approach utilizing deep deterministic policy gradient is proposed with the aim of minimizing the total cost associated with data transmission delay and energy consumption.
	Centralized [154]	MEC	Accelerated convergence	Absence of a specified mobility model	Proposed a DRL algorithm aimed at minimizing both execution delay and energy consumption.
	Distributed [155]	MEC	Optimization of offloading decisions and resource allocation considering latency and energy constraints.	Energy feasibility of switching between communication modes may be challenging.	Introduced a decentralized algorithm that offers an optimal solution for the problem of computation efficiency. The algorithm is designed to jointly optimize task offloading decisions and computation resource allocation.
	Distributed [156]	MEC	Attains convergence within a restricted number of iterations,	Inadequate performance observed in highly complex vehicular environments	Developed a Double Deep Q-Network (DDQN) algorithm for predicting the offloading behavior of User Equipment

(continued on next page)

Table 6 (*continued*)

Object	Scheme	Computing Architecture	Advantages	Disadvantages	Proposed solution
			thereby mitigating transmission overhead.		(UEs) with tasks having a semi-online distribution. The algorithm calculates and updates the total rewards after each offloading decision.
Minimization of Latency, Communication, and cost of Computation	Distributed [157]	MEC	Valid system model.	Unrealistic assumption regarding the initial offloading tasks in the vehicular route.	Proposed a contract-based computation offloading approach to enhance both the benefits of MEC service providers and the utility of vehicles.
Object	**Scheme**	**Computing Architecture**	**Advantages**	**Disadvantages**	**Proposed solution**
Minimization of Latency, Communication, and Cost of Computation	Distributed [158]	MEC	An economically efficient approach that considers deadline constraints.	Insufficient organization.	A novel approach has been developed to improve the efficiency of offloading in vehicles by considering deadlines and cost-effectiveness as key factors.
	Centralized [159]	MEC	Effectively manages interdependent tasks.	Does not take into account the specific use cases and variations in vehicular mobility patterns	To address the need for fast offloading in vehicular environments, an offloading algorithm called SVMO was developed. SVMO draws inspiration from the principles of Support Vector Machines (SVM) to optimize the offloading process and improve efficiency.
	Centralized [160]	MEC	Resource allocation optimization for efficient offloading	Lacks analysis of mobility characteristics	Developed the JOPRAO algorithm, which takes into account the offloading proportion, communication resource, and computation resource allocation.
	N/A [52]	MEC	Demonstrated the system's validity through the utilization of real-world traces.	Lacks clarity and organization.	Proposed heuristic offloading for efficient placement and scheduling of vehicular application components between OBU and the cloud.
	Distributed [161]	MEC	Efficient resource utilization/ Simplified approach.	Does not meet the requirements for ensuring QoS in a realistic manner.	Introduced a decentralized offloading algorithm utilizing game theory principles within vehicular edge networks.
	N/A [162]	MEC	An effective task and resource allocation model is proposed to optimize the allocation of tasks and resources in the system.	A large number of contracts and the lack of latency analysis are considered in the proposed approach.	An algorithm for server selection and resource allocation based on a contract theoretic approach is developed to optimize the offloading scheme for the service provider.
	N/A [163]	MEC	Scalability, practicality, and offers improved incentives for vehicles, thereby promoting their active involvement and motivation.	Security vulnerabilities arise when sharing resources between MEC platforms and vehicles, leading to insecure data access and potential breaches.	A price-based two-stage Stackelberg game is utilized to model and simulate the computation trading process between Cloudlet and vehicles, allowing for efficient resource allocation and pricing strategies.
	N/A [164]	MEC	Efficient computational complexity	Inadequate resource allocation in densely populated areas.	An optimal approach is presented for task offloading and resource allocation in both independent and cooperative MEC servers. This ensures efficient decision-making and maximizes the overall performance of the system.
	Centralized [165]	MEC	Well-structured system model	Insufficient coordination, inadequate management	A novel algorithm is introduced that enables dynamic organization of computing resources through multi-platform offloading and resource allocation.
	Centralized [166]	MEC	Three distinct mobility models and road conditions are employed for analysis and evaluation.	prohibiting partial offloading at the local vehicle.	Proposed pricing-based algorithms for one-to-one and one-to-many matching in computation offloading.
	N/A [167]	MEC	Well-structured problem formulation.	Vehicular environment with minimal complexity.	Presented an optimization algorithm for selecting, computing, and adjusting the contention window, resulting in a reduction in the execution completion time.
	Centralized [168]	MEC	Collaborative computing model effectively reduces computational latency and improves service reliability.	Significant computational complexity.	A collaborative computing model is proposed to address the issue of computing latency and improve service reliability in vehicular networks.
	N/A [169]	MEC	Well-suited for multi-user and multi-server environments.	Does not account for the variability in tasks' parameters and MEC network.	The presented algorithm combines offloading decision and task scheduling to minimize task delay and computing resource consumption.

(*continued on next page*)

Table 6 (*continued*)

Object	Scheme	Computing Architecture	Advantages	Disadvantages	Proposed solution
	Distributed [170]	MEC	Optimal offloading decisions that take into account both latency and energy considerations.	without specific limitations and mobility scenarios.	A comprehensive integration model is formulated to combine computational offloading and resource allocation, with the objective of minimizing the overall system cost in terms of latency and energy consumption.
	Distributed [171]	MEC	A well-defined and comprehensive system model.	Lack of specificity in the considered application domains.	Introduced a cost-effective offloading model tailored for 5G-enabled vehicular networks, aiming to minimize the overall offloading cost while ensuring adherence to latency constraints.
	Distributed [172]	MEC	Well-defined model.	Not tailored to specific applications and lacks consideration of mobility factors.	Designed a multilevel offloading approach based on Stackelberg game theory, aiming to maximize the utilities of both vehicles and VEC servers.

unpredictable nature of vehicular mobility.

In response to the challenges posed by the dynamic movement of vehicles, recent research has witnessed the emergence of solutions that leverage cutting-edge technologies. EC, ML algorithms [203], and adaptive decision-making approaches [204] have been explored to enhance the efficiency of task offloading in dynamic Edge-IoV scenarios [205–207]. These solutions contribute to the development of a robust offloading framework capable of dynamically adapting to the ever-changing vehicular environment. Fig. 14 depicts the ComOf scenario for networks characterized by both high and low density.

The process of computation offloading unfolds through the following steps. Vehicles possess a specific computational capacity that guides their decisions regarding task offloading. This underscores the online, distributed, and low-complexity nature of the computation offloading scheduling algorithm. Each vehicle can obtain necessary data for offloading from the RSU. If a vehicle opts not to offload its computation task, the On-Board Computers (OBCs) execute the task. Conversely, if a vehicle chooses to offload, it generates a packet containing computation data and relevant task attributes, sending this packet to the RSU via V2I communication. Upon receiving the computation offloading request, the edge server deliberates whether to execute the task locally or transmit the packet to the cloud server. Upon confirmation by the cloud server or edge servers, if there is no VM for the vehicle's application task, one is created with appropriate computation resources; otherwise, the task is added to the task queue of the corresponding VM. Upon completion, the results are transmitted back to the vehicles, as illustrated in Fig. 15.

7.1. Task offloading optimization

Existing studies on computational offloading typically center on enhancing both delay and energy efficiency. In VEC, diverse approaches and techniques are applied to address various goals within the realm of computational offloading. Specifically, the categorization of computational offloading in VEC reveals two distinct divisions, as illustrated in Fig. 16.

i. User-Side Optimization:

- Delay minimization: The advancement of vehicle applications has significantly enhanced user experiences. However, these applications are often characterized by real-time demands, complexity, and intensity [208]. Prolonged processing time for application tasks may compromise data effectiveness, potentially leading to traffic accidents. VEC delay encompasses transmission, computing, and communication delays. Computing delays involve offloading, queuing, and processing delays. Consequently, the reduction of latency in VEC applications holds paramount importance. Zhao et al. [209] introduced a scheme for edge cache and computing management, optimizing service caching, request scheduling, and resource allocation strategies. Results indicate suboptimal delay performance with this approach.
- Delay and Energy Consumption Optimization: Deploying computing-intensive applications in VEC with strict time constraints poses challenges, often leading to increased energy consumption. When a vehicle's energy is low, meeting onboard application requirements becomes challenging. Thus, it's crucial to consider energy consumption alongside time constraints, necessitating a VEC offloading approach that balances and minimizes both delay and energy usage. This optimization aims to achieve higher channel gain and reduce local calculation energy consumption. Zhan et al. [154] argued an adaptive learning-based task offloading algorithm that comprehends neighboring vehicles' delay performance during offloading calculations. This algorithm effectively minimizes the average load delay and energy consumption for each task vehicle. In a related context, Zhan et al. [210] delved into the ComOf scheduling issue in VEC scenarios. They modeled it using a well-crafted Markov decision process and developed a cutting-edge, near-end policy optimization algorithm based on DRL.
- Delay and Cost Optimization: VEC strives to minimize offloading, encompassing communication and computing costs. Illustrated through cellular network communication, vehicles may incur charges for data transmission in a cooperative distributed computing framework, wherein vehicles function as computing resources for VEC. Qiao et al. [175] presented a cooperative task offloading and transmission mechanism, effectively mitigating system delay and energy consumption. Meanwhile, Zhao et al. [147] discussed a cooperative ComOf and resource allocation optimization approach, featuring a designed algorithm for distributed ComOf and resource allocation. This method enhances overall system utility, considering factors like task processing delay and computational resource costs, achieved through the optimization of offloading strategy and resource allocation using game theory principles. Salman Raza and colleagues [181] explored cloud-based a motion-aware partial task offloading algorithm aimed at minimizing the overall offloading cost. This approach considers the expenses associated with necessary communication and computing resources. It involves dividing the task into three segments and determining the allocation ratio for each part based on the vehicle's available resources. Simulation results demonstrate the effectiveness of this method in reducing communication costs for proximate vehicles and alleviating the workload on VEC servers, particularly in densely populated urban settings.

ii. System Server Optimization:

- System utility maximization: In the context of ComOf for VEC, maintaining a balanced distribution of system resources is essential. The goal is to prevent devices from being overloaded or remaining idle, thus optimizing the overall system utility of VEC. Liu and

Table 7
Overview of V2X computation offloading schemes.

Object	Scheme	Computing Architecture	Advantages	Disadvantages	Proposed solution
Minimization of Latency	Both [173]	MEC + NV	Real-world vehicular mobility patterns	Lack of clarity and structure in the presentation.	A framework is proposed to optimize task offloading and resource utilization by identifying suitable surrogates for dynamically allocating tasks. The aim is to enhance efficiency and maximize the utilization of available resources.
	Centralized [174]	MEC + VCC	Well-defined formulation	Uniform treatment of tasks regardless of their deadlines.	Suggested the BETA policy as a solution to address the challenge of complexity, where tasks are assigned a minimum number of replicas.
	Distributed [175]	MEC + NV	Load Balancing	The system experiences high latency when there is a limited number of resource-rich vehicles available.	The paper introduces the VE-MAN hierarchical network framework and proposes a hybrid control collaborative task offloading scheme to eliminate redundant computation tasks.
	Distributed [176]	MEC + NV	A well-defined system model that accurately captures the network characteristics and includes practical estimation methods for transmission rates.	Insufficient analysis of challenging road conditions and their impact on the proposed federated offloading scheme.	The proposed scheme involves a federated offloading approach that combines V2I and V2V communication in MEC enabled vehicular networks.
	Centralized [81]	MEC + NV	The organization of the system is well-structured, presenting the concepts and findings in a logical and coherent manner	In complex and dynamic IoV environments, it is important to consider not only performance optimization but also stability to ensure consistent and reliable system operation.	investigated an integrated architecture that combines satellite networks and 5G IoV to leverage the benefits of both technologies.
	Distributed [177]	MEC or NV	A comprehensive system model that considers various factors such as vehicle mobility, network connectivity, and resource availability.	Lack of analysis of dynamic vehicle mobility patterns.	A novel algorithm is developed to determine the optimal offloading route for distributing computing tasks between a source vehicle and a target vehicle.
	Distributed [178]	MEC + NV	Achieving rapid convergence while avoiding premature convergence is a key focus of the proposed approach.	Applicable across various domains, without being limited to specific applications.	A KD (Knowledge Distillation) service-offloading framework is proposed to derive an optimal service offloading policy that achieves long-term performance optimization.
	Distributed [179]	MEC + VU	Efficient utilization of vehicular communication resources.	Not focus on a specific application and does not address the issue of delay performance.	A system called Mobile Edge is introduced, which leverages the computational power of passengers' mobile devices to enhance the capabilities of the on-board Vehicle Control Unit (VCU).
	Distributed [180]	MEC + NV	The formulation used in the study is considered acceptable and provides a solid foundation for addressing the research problem.	Does not take into account any specific mobility model.	The recommended architecture is based on SDN and aims to enable low-latency computing services in vehicular and RSU infrastructures.
	Centralized [111]	MEC + NV	The proposed approach includes a practical mobility analysis to assess the movement patterns and dynamics of vehicles in real-world scenarios.	The presence of a large number of offloading hops can result in significant delays in the system.	The proposed strategy is a multi-hop VANETs-assisted offloading approach that leverages the link correlation theory to optimize task offloading decisions and resource allocation.
	Distributed [181]	MEC + NV	Effectively utilizing the available vehicular resources and accurately estimating the transmission rates based on practical scenarios.	primarily focuses on independent tasks, neglecting the consideration of interdependent task dependencies.	The MAP task offloading algorithm is designed to allocate tasks among the local, V2V, and V2I components, taking into account the task allocation ratio. Additionally, the transmission rates for V2V and V2I communication are empirically measured based on practical assumptions.
	Distributed [182]	MEC + NV	An effective system model that captures the essential aspects of the problem and provides a realistic representation.	Did not address the handover challenges associated with computation offloading.	Introduced a strategy based on Q-Learning to minimize the average delay in computation.
Object	**Scheme**	**Computing Architecture**	**advantages**	**disadvantages**	**Proposed solution**
Incentive-Based Resource Allocation, Load Balancing, and Server Selection	N/A [183]	MEC + NV	Cost-effective deployment	Lacks in addressing security and resource management challenges due to its self-organized and ad-hoc nature.	A VFC system is proposed with the aim of augmenting the available resources and enhancing the achievable capacities.
	N/A [184]	MEC + NV	local and FC resources and incorporates an adaptive transmission mode.	multiple offloading hops can result in notable delays.	A comprehensive offloading scheme is proposed that enables the efficient transfer of tasks to the MEC server using

(continued on next page)

Table 7 (*continued*)

Object	Scheme	Computing Architecture	advantages	disadvantages	Proposed solution
					either direct transmission or predictive relay transmission.
	Distributed [185]	MEC + NV	Efficient and minimal delay and overhead.	Limited resources in high-density areas.	Introduced a distributed algorithm for optimal computation offloading decisions in vehicular terminals.
	Centralized [186]	MEC or NV	Designed an offloading strategy that takes into account the characteristics of specific applications and focuses on maximizing long-term rewards.	Lacks comprehensive analysis of latency and its impact on the proposed framework.	Presented a multi-timescale framework based on DRL to address challenges related to communication, caching, and computing.
	Centralized [35]	MEC + NV	Optimal trade-off between service latency and quality degradation.	Lack of incorporation of multi-vehicular cloudlets, leading to limited scalability and potential inefficiencies in resource utilization.	Developed a dynamic task allocation scheme that strikes a balance between service latency and quality loss, ensuring optimal performance.
	Distributed [187]	MEC + NV	Effectively addresses resource allocation and load balancing challenges.	Underutilization of vehicles as fog nodes in the proposed system architecture.	Developed a distributed traffic management system that utilizes vehicle-based fog nodes, with the modeling of these nodes based on queuing theory.
	N/A [188]	MEC + NV	Alleviates the burden on MEC systems and offers cost-effective deployment.	Insufficient resource management and security considerations.	Introduced a load-aware scheme for task offloading that takes into consideration the load balance of MEC servers and cost prediction.
	Distributed [63]	MEC + NV	Provide incentives to both edge servers and cloud servers.	Limited resource availability in high-density areas.	Introduced a contract-based incentive model leveraging contract theory to maximize the utility of BS. Additionally, devised a stable matching algorithm based on pricing to ensure stability in the matching process.
Optimizing Energy Efficiency	Centralized [189]	MEC + NV	The vehicle fog node efficiently serves multiple tasks or vehicles, maximizing resource utilization and enhancing overall system performance.	Resource scarcity in high-density areas poses a challenge in terms of providing sufficient resources to meet the demands of multiple users.	Two online strategies were proposed to examine the task scheduling and offloading decision in scenarios involving multiple smart vehicles.
	Centralized [190]	MEC + NV	Manageable complexity.	Not tailored to a particular application context.	Proposed an online algorithm that combines Lyapunov's method and SCA-based optimization to minimize delays while satisfying energy constraints.
	Distributed [191]	MEC + NV	Enhanced QoE through various improvements and optimizations.	Lacks consideration for specific applications, limiting its applicability and effectiveness in addressing domain-specific challenges.	Proposed a DRL algorithm to optimize the QoE by minimizing energy consumption and achieving optimal performance.
Minimization of Cost	Distributed [192]	MEC+NV	Application-aware offloading and long-term reward.	Lacks comprehensive analysis of latency.	Provided is a probability-based contact graph, a dynamic offloading tree, and the use of a greedy algorithm, collectively forming a comprehensive approach to address spatiotemporal constraints.
	Centralized [193]	MEC+NV	Reduce the cost associated with transferring data and utilizing computational resources for task offloading.	The presence of multiple offloading hops has the potential to result in considerable delays.	Suggested design underwent validation through a practical application scenario, and the acquired outcomes demonstrated its alignment with specific application requirements, along with ensuring commendable scalability and responsiveness.
	Distributed [194]	MEC+NV	Allocation of resources between a PV system and a service provider.	High complexity and slow learning rate for large problems and high reliance on certain factors.	Proposed Knowledge-Driven (KD) service offloading decision framework. This framework utilizes DRL to formulate the offloading decision of multi-tasks in a service as a long-term planning problem.
	Distributed [195]	MEC+NV	Improving QoE, Load Balancing	Overload of fractional Edge Nodes (ENs) due to the explosive growth of offloaded vehicle applications.	Introduced solution is computation offloading method, designed to address the challenges of offloading applications in overloaded ENs to other idle ENs. This method jointly optimizes to reduce application offloading delay and offloading cost across ENs while achieving the load balance of ENs globally.
	Distributed [196]	MEC+NV	Revenue maximization Through the development of an appropriate pricing algorithm	Limited practical validation	A novel optimization methodology is introduced, emphasizing a cost-based approach that simultaneously evaluates the expense associated with partial offloading in comparison to the pricing structure of the MEC server. The

(*continued on next page*)

Table 7 (*continued*)

Object	Scheme	Computing Architecture	advantages	disadvantages	Proposed solution
					proposed methodology is subjected to performance analysis.

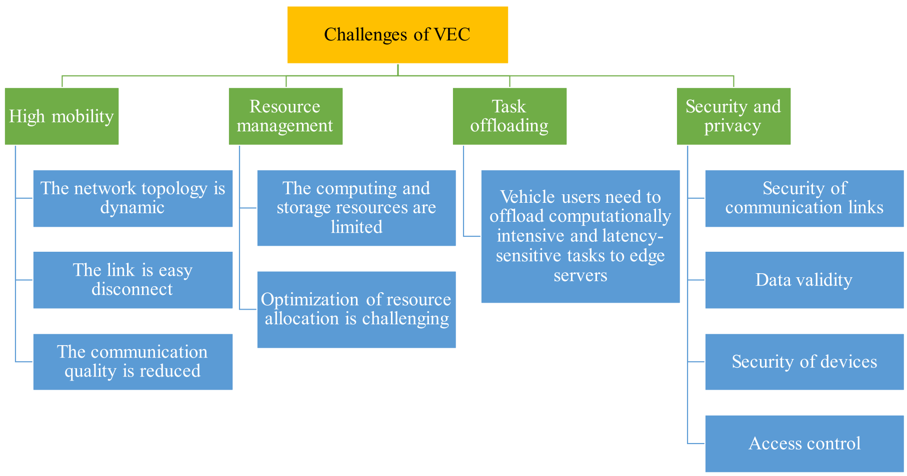

Fig. 13. Challenges of VEC.

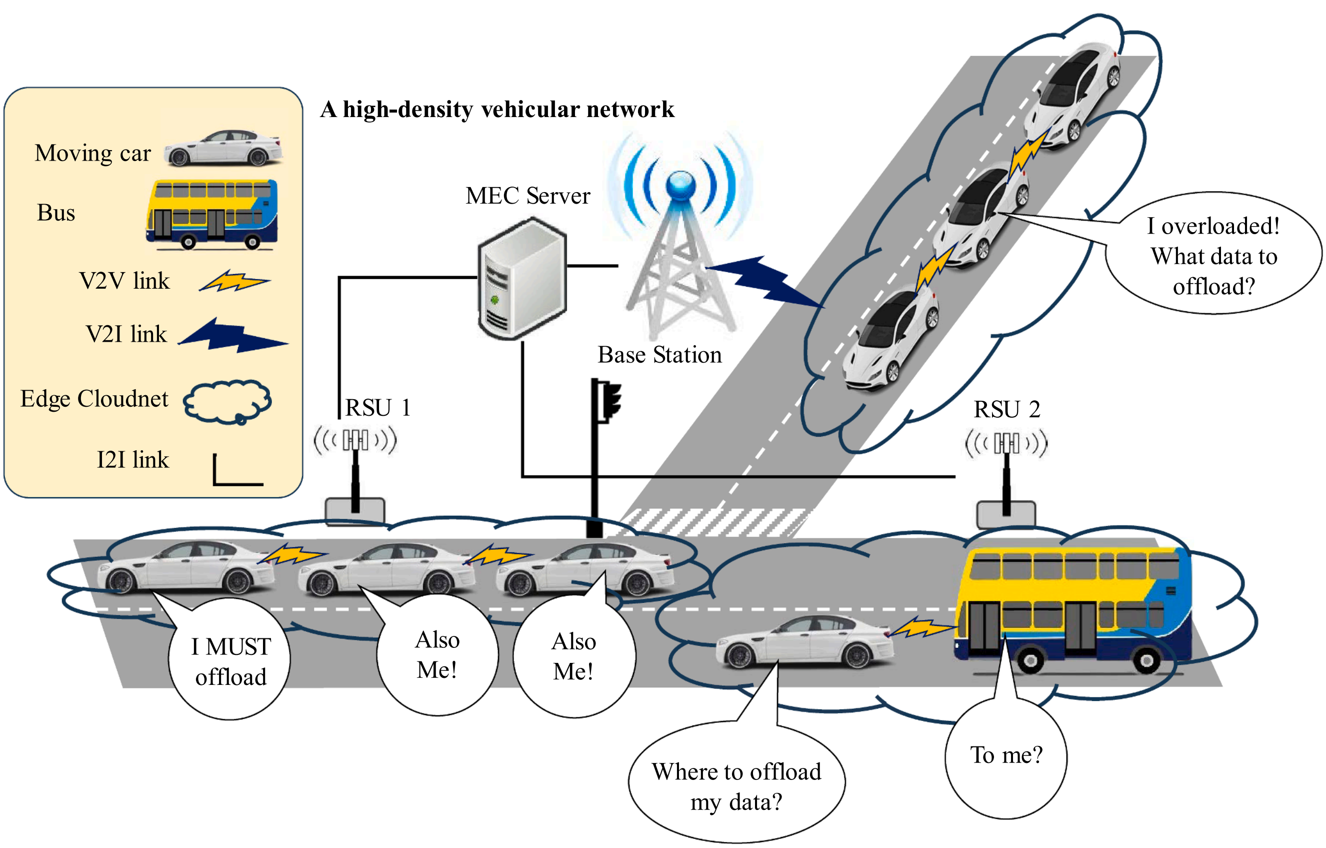

Fig. 14. ComOf scenario.

collaborators [211] addressed this challenge by proposing an offloading strategy and resource allocation scheme tailored for both vehicle edge servers and fixed edge servers. To address the inherent randomness and uncertainty in vehicle communication, they transformed the total utility maximization problem of the VEC network into a semi-Markov process. They provided a reinforcement learning

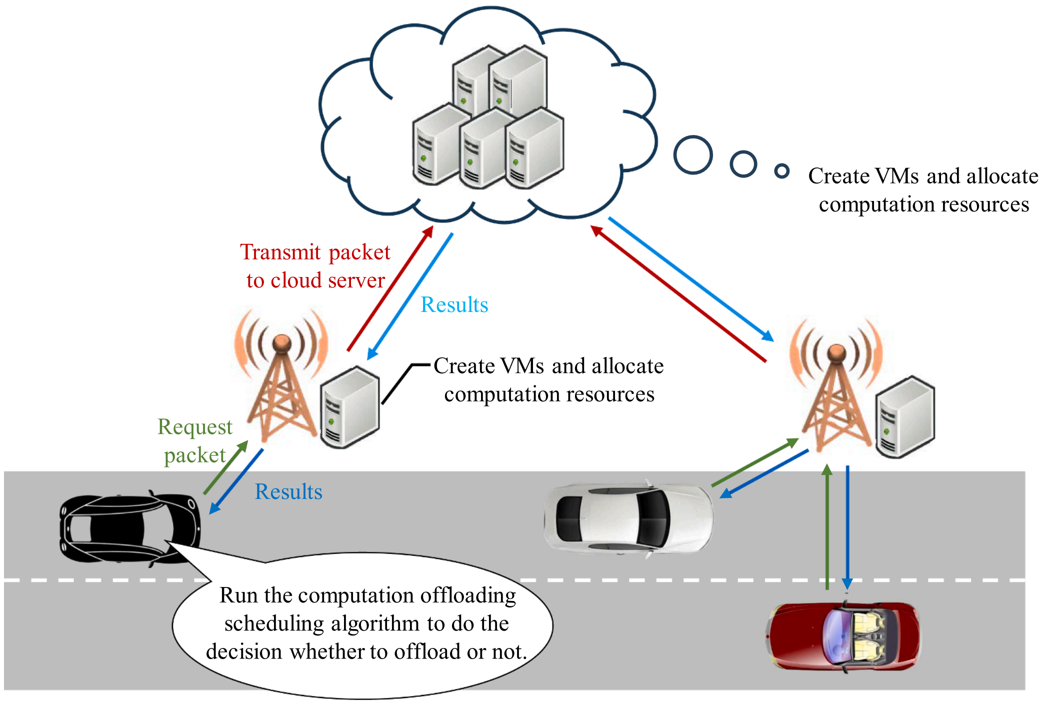

Fig. 15. ComOf process.

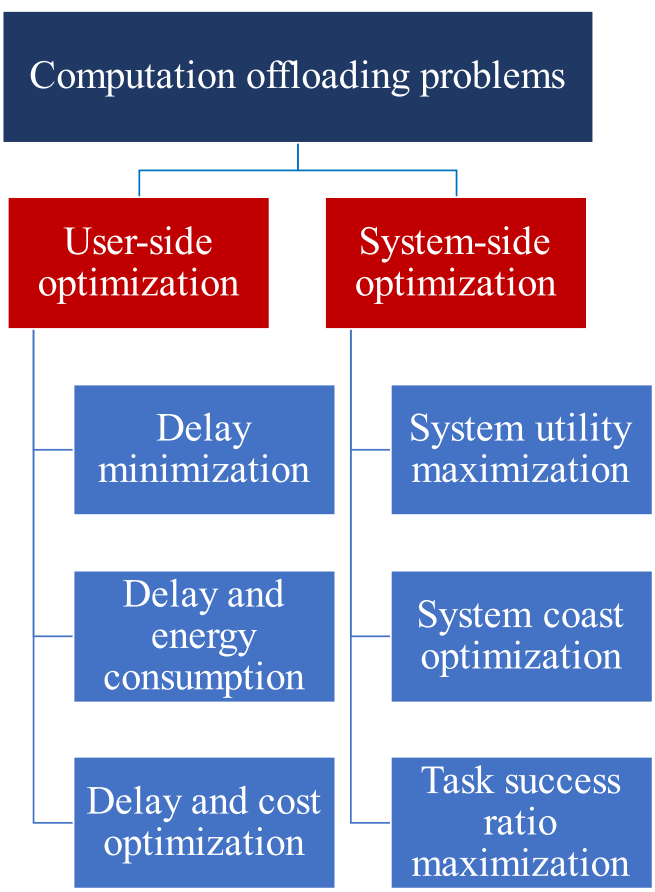

Fig. 16. ComOf optimization in VEC system.

method and a DRL approach based on Q-Learning to identify optimal strategies for ComOf and resource allocation. Additionally, Dai et al. [212] examined an Artificial Intelligence (AI)-enhanced VEC and cache scheme. This architecture combines EC with intelligent resource caching, enabling cross-layer offloading, multi-point caching, and delivery. The integration of DRL and Deep Deterministic Policy Gradients (DDPG) algorithms enhances system utilities.

- System cost minimization: Tan et al. [186] examined a comprehensive resource allocation framework encompassing communication, caching, and computing within VEC networks. Addressing the optimization challenge of resource allocation, they put forth a multi-time scale framework-based algorithm rooted in DRL. This approach aims to optimize resource allocation while contending with constraints such as limited storage capacity, computational resources, and stringent delay requirements for both vehicles and RSUs. In a related effort, Zhang et al. [213] devised a strategy for joint cache and computing allocation to minimize system caching. They investigated mobile recognition active cache technology, emphasizing acquiring and storing video content in the base station's cache. The strategy employs a K-Nearest Neighbors (KNN) algorithm to identify the optimal joint view set, maximizing total rewards and minimizing system costs.
- Mission Success Rate Maximization: Qiao et al. [214] concentrated on collaborative edge caching and supplementary caching, presenting a cooperative edge caching strategy to enhance content placement and delivery. The optimization challenge involves a two-time scale Markov decision process. Liang et al. [215] utilized model resource sharing and employed deep Q-network-based approaches as multi-agent reinforcement learning tools in network spectrum sharing. They designed a V2V spectrum and power allocation scheme to enhance the reliability of payload delivery over the V2V link and achieve periodic sharing of safety–critical messages.

8. VEC technical categorization

8.1. Opportunities

The key advantages of VEC can be listed as:

i. **Response Time:** The response time includes the delivery time of the data offloaded to edge servers and back and the time for their processing in the servers. Compared with the clouds, the edge servers are situated nearer to vehicular users, significantly lowering the execution time, which is specifically advantageous for delay-sensitive applications, like safety applications.
ii. **Energy Efficiency:** The growing number of smart vehicles has led to an explosive boost of diverse vehicular applications which will require a huge deal of energy. With the help of VEC, electric vehicles with limited energy sources can receive sufficient support for such applications.
iii. **Bandwidth:** The rapid proliferation of smart vehicles leads to a substantial increase in data generation encompassing diverse content requests. However, centralized management through CC may struggle to meet the bandwidth demands for processing such massive amounts of data, especially considering the long distances involved with user interactions. VEC presents an effective solution to address this bandwidth challenge by leveraging backhaul networks and relocating computation and storage resources from the cloud to the network edge. This approach enables data processing and storage tasks to be performed closer to the source, reducing the burden on centralized infrastructure, and optimizing bandwidth utilization. By distributing computational resources to the edge, VEC enhances the efficiency and capacity of the overall network, ensuring that smart vehicles can access the required bandwidth for seamless data processing and communication.
iv. **Storage:** Unlike the case for clouds, VEC enables data storage in edge servers located near vehicular users. The caching technology also provides the possibility of in-time access to the stored data, while decreasing the storage loads of the remote clouds.
v. **Proximity Services:** Diverse proximity services can be offered due to the closer distance of the servers to vehicular users in VEC. This can improve the user experience while efficiently managing the traffic. For instance, upon arrival at the site and uploading sensing information from vehicles, edge servers can contribute to data processing and construct a High Definition (HD) map and send it to vehicles.
vi. **Context Information:** In VEC, edge servers can attain real-time information on the behavior and location of vehicles, traffic status, and network conditions which can be utilized to improve various applications. For instance, such real-time data can be employed for content delivery to vehicular users according to their interests.

8.2. Obstacles

Some of the technical issues of VEC are categorically discussed as listed in Table. 8.

i. **Latency:** Many burgeoning vehicular applications require real-time mobility support (e.g., positioning systems and smart traffic lights which may result in network latency. The latency cannot be assigned only to long cloud-vehicle distance as it could be due to the routing inability or delayed queueing as well. Nonetheless, new vehicular applications require a large computational capacity for processing complex tasks. Several approaches have been developed to lower the data transmission delay while maximizing the throughput.

- Routing Approach: The routing approach applies geographic routing (position-based) for local decision-making. A node transmitting a data packet should consider three positions: its current position, destination, and its one-hop neighbor. A routing scheme was proposed under the title of Improved Geographic Routing (IGR) [216], targeting the vehicles moving in a city environment. IGR exploits VEC for utilizing computational

Table 8
VEC technical issue.

Issue		Reference	Year	Paper	Contribution
Latency	**Routing**	[216]	2018	FC enables geographic routing for urban area vehicular network	Proposing an enhanced geographic routing (IGR) scheme for routing in urban environments, specifically tailored to vehicles in motion.
		[217]	2021	Edge Network Routing Protocol Base on Target Tracking Scenario	Introducing a routing and forwarding protocol for edge networks specifically designed for target tracking scenarios.
	SDN	[47]	2017	Latency Control in Software-Defined Mobile-Edge Vehicular Networking	Introducing a comprehensive suite of latency control mechanisms that encompass various aspects such as radio access steering and cache processing at the base station.
		[218]	2019	Detour: Dynamic Task Offloading in Software-Defined Fog for IoT Applications	Addressing the problem of task offloading in a software-defined access network in which multi-hop IoT access-points (APs) connect IoT devices to FC nodes.
	5G	[219]	2020	Secure and Efficient Privacy-Preserving Authentication Scheme for 5G Software Defined Vehicular Networks	Using an RL instead of a certificate revocation list (CRL) to decline the verification delays due to checking the long CRLs and the storage costs arising from a large number of pseudonyms.
		[220]	2017	Foud: Integrating Fog and Cloud for 5G-Enabled V2G Networks	5G technologies were employed to resolve the problem of the explosive growth of vehicular terminals and mobile data traffic.
Scheduling and Load-Balancing		[221]	2017	Exploring fog computing-based adaptive vehicular data scheduling policies through a	Introduction of a scheduling scheme based on queue length and response time and formulation of a

(continued on next page)

Table 8 (*continued*)

Issue	Reference	Year	Paper	Contribution
			compositional formal method – PEPA	design for a vehicular cloud, based on a compositional approach (PEPA).
	[222]	2022	Enhanced time-constraint aware tasks scheduling mechanism based on predictive optimization for efficient load balancing in smart manufacturing	An enhanced Time Constrain Aware (TCA) tasks scheduling mechanism was proposed as an improved version of Fair Emergency First (FEF) scheduling which takes accurate decision (prediction) measures and minimal task time into account.
	[223]	2022	SDN-based load balancing technique in the internet of the vehicle using integrated whale optimization method SDN Based Load Balancing Technique in the Internet of Vehicle using Integrated Whale Optimization Method	Introduction of a software-defined network-based load balancing strategy for IoV to minimize latency and tasks with the help of cloud and EC devices using an integrated whale optimization algorithm (WOA).
Offloading	[184]	2017	Predictive offloading in cloud-driven vehicles: using mobile-EC for a promising network paradigm	Development of a computational offloading infrastructure with emphasis on the computational efficiency of the transfer frameworks of V2I and V2V communication modes.
	[224]	2020	Resource Allocation for Vehicular FC using Reinforcement Learning Combined with Heuristic Information	Presenting an offloading scheme based on deep learning neural networks to assess the movement patterns of vehicles and forecast the accessibility of resources for subsequent offloading decisions. In this method, the Proximal Policy Optimization (PPO) algorithm was employed by integration of the Recurrent Neural Network (RNN) with a Deep Neural Network (DNN).
	[38]	2020	Blockchain and Learning-Based Secure and Intelligent Task Offloading for Vehicular	Development of an online learning-based framework for intelligent task offloading where vehicles are trained to discover the optimal task offloading strategy with the minimum latency. The learning process relies on handover cost, queuing delay, and the reliability of the accessible fog nodes
Resource Management	[226]	2022	A Federated Deep Learning Empowered Resource Management Method to Optimize 5G and 6G Quality of Services (QoS Intelligent Transportation System	Providing insight on resource allocation in 5G vehicular networks. Furthermore, a new federated deep reinforcement learning (FDRL) was presented.
	[227]	2022	An Intelligent Approach for Cloud-Fog-Edge Computing SDN-VANETs Based on Fuzzy Logic: Effect of Different Parameters on Coordination and Management of Resources	Using an integrated fuzzy logic system to determine the best resources for vehicles under different conditions. These conditions include the quality of the intravehicular networks, their size and longevity, the number of accessible resources, and the application requirements.
	[228]	2017	Resource Management in Fog-Enhanced Radio Access Network to Support Real-Time Vehicular Services	The management strategies of each edge node (fog node) in FeRANs were targeted to improve resource management of channels. This focused on

(*continued on next page*)

Table 8 (*continued*)

Issue	Reference	Year	Paper	Contribution
				enhancing QoS for real-time vehicular services. To achieve this, fog resource reservation and fog resource reallocation mechanisms were developed. This approach enhanced the on-hop probability of real-time vehicular services, even in situations where fog resources were heavily utilized.
Security and Privacy	[229]	2017	A Privacy-Preserving Vehicular Crowdsensing-Based Road Surface Condition Monitoring System Using Fog Computing	Proposing a privacy-preserving protocol that enhances security in a crowdsensing-based road condition monitoring system by leveraging EC. The scheme utilizes certificateless aggregate signcryption schemes to reduce communication overhead and expedite the verification process, thereby improving overall efficiency.
	[230]	2018	Secure, efficient, and revocable data sharing scheme for vehicular fogs	Introducing a data-sharing scheme that employs effective decryption techniques in a multiauthority CP-ABE system, ensuring protection against collusion attacks.
	[231]	2017	A Secure Trust Model Based on Fuzzy Logic in Vehicular Ad Hoc Networks With Fog Computing	Presenting a fuzzy trust model that efficiently identifies faulty nodes and unauthorized intruders while effectively handling data uncertainty in vehicular networks.

resources and vehicular communications. In its updated greedy forwarding mode, the link error rate is treated in the path selection. The IGR remarkably improved the packet rate along with end-to-end delay.

An edge network routing and forwarding protocol was also developed on the basis of target tracking scenarios [217]. Such a protocol can cover the dynamic variations in node locations and the elastic node scale expansion. Failure of an individual node will not influence the overall network while ensuring efficient real-time communication with less overhead. According to experimental results, the protocol managed to decrease the communications volume of the edge network, enhance network efficiency, and decide on the best sample interval to minimize the network delay.

- SDN approach: To ensure robust resource management and effective traffic control, this networking approach emphasizes centralized control of the logical network. SDN is at the core of this framework, offering network programmability, flexibility, and knowledge. In the context of wireless services, addressing high-speed vehicular environments, the most critical QoS requirement is minimizing delay. Software-Defined Mobile-Edge Vehicular Networking tackles this challenge by employing a comprehensive set of latency control strategies [47]. These strategies range from radio access steering to cache processing at base stations, all aimed at meeting the stringent delay requirements. However, when deploying vehicular technologies for autonomous driving systems, it becomes crucial to address the issue of advanced processing capabilities. Wireless infrastructures alone cannot guarantee the safety of drivers in such scenarios. Therefore, sophisticated processing capabilities need to be integrated into the system to ensure the reliability and security of the driving machines.

Misra and Saha [218] addressed the problem of task offloading in a software-defined access network in which multi-hop IoT Access Points (APs) connect IoT devices to FC nodes. The developed scheme considered the following aspects: a) optimum decision on local or remote task computing, b) selecting optimal fog node, and c) choosing an optimal offloading path. The authors thus formulated a multi-hop task offloading problem in the form of an Integer Linear Program (ILP). Regarding the non-convexity of the feasible set, a greedy-heuristic-based approach was developed. The greedy solution considered multi-hop paths, energy consumption, delay, and dynamic network conditions (e.g. link utilization).

- 5G Approach: The emergence of 5G mobile communication networks not only improved the performance of the current vehicular networks but also supported new applications in vehicular networks. A novel architecture of 5G software was examined [219] to develop a safe and privacy-preserving authentication scheme for vehicular networks. The mentioned scheme used elliptic-curve public-key cryptography and a registration list for achieving efficient message authentication and avoiding the use of growing certificate revocation lists. This scheme employed an RL rather than a Certificate Revocation List (CRL) which can decrement the verification delay arising from the evaluation of the long CRL and the storage cost due to a large number of pseudonyms in the CRL.

A new hybrid-computing model architecture (Foud) was also explored [220] for V2G networks. As suggested by its name, Foud supplies edge/fog and cloud to the V2G networks. The infrastructure of Foud includes temporary fog and permanent cloud sub-models. EC can be applied as a Foud sub-model because of its dynamic mobile communication resource. Additionally, 5G technologies can overcome the issue of the increase in vehicular terminals and mobile data traffic.

ii. **Scheduling and Load-Balancing:** Vehicle networks provide efficient communication to enhance data dissemination among vehicles. Many vehicles conduct data dissemination, enhancing the load. The latest scheduling algorithms can adapt to the diverse challenges of queue length. The classic shortest queue policy is one of these algorithms. The shortest queue does not necessarily mean minimal waiting time; therefore, time-based scheduling schemes are more effective and reliable. A scheduling method was developed by Chen et al. [221] based on response time and queue length to enable the possession of varying communication environments for vehicular communications. They also established a three-layer vehicular cloud based on edge/fog computing. The developed architecture relies on a compositional approach (PEPA) which helps in modeling large-scale systems owing to its compositionality and abstraction.

An enhanced Time Constrain Aware (TCA) tasks scheduling mechanism was also developed [222] as an improved version of the Fair Emergency First (FEF) scheduling, considering accurate decision (prediction) measures and minimal task time for efficient task scheduling. This study was aimed at efficient task execution sequence to enhance smart manufacturing and efficiency of resource utilization in real-time through maximization of the usage of smart machines, minimization of tasks idle time, and autonomous control of the smart manufacturing environment using sensors and actuators.

Darade and colleagues [223] proposed a software-defined network-based load balancing strategy for IoV which minimized the delay and tasks of the IoV by cloud and EC devices through an integrated Whale Optimization Algorithm (WOA). The terminal users produce high traffic, explaining the enhanced latency of the cloud network. Owing to software-defined networking, clouds, IoV, and fog networking are benefited from centralized control and global knowledge. The performance of the proposed model was compared with the WOA in terms of latency. Based on experimental findings, the IoV-based load balancing of SDN through integrated WOA outperformed other load balancing schemes. It could effectively minimize the latency while improving the QoS in fog computing, providing mobility and position awareness in IoT.

iii. **Offloading:** Thanks to their proximity to vehicular users, edge servers can lower the transmission cost and provide a rapid response in the offloading services. Despite their prompt response, edge servers usually encounter resource limitations in comparison with conventional cloud servers with large computational capacities. The edge servers require a certain time for computation tasks. This finds especial importance in the edge servers situated at the road segments, as they face a large density of vehicles. Zhang et al. [184] examined a computational offloading infrastructure with emphasis on the computational efficiency of the transfer frameworks of V2I and V2V communication. They also proposed an efficient predictive combination-mode relegation scheme considering the execution time of the tasks and vehicle mobility. In the mentioned model, the tasks are offloaded to the MEC servers through direct upload and predictive relay transmissions.

Lee and Lee [224] developed an offloading scheme based on deep learning neural networks for evaluating the mobility patterns of various vehicles and predicting resource accessibility for future offload decisions. They employed the Proximal Policy Optimization (PPO) algorithm as the DRL method through the integration of the Recurrent Neural Network (RNN) with the Deep Neural Network (DNN). This model was then applied to assess previous resource allocation trends within the VFC environments. Liao and coworkers [38] developed an online learning-based framework for intelligent task offloading. In their model, vehicles were trained to find an optimal task offloading strategy with minimal latency. The training process relies on handover costs, queuing delays, and the reliability of the accessible vehicular fog nodes.

iv. **Resource Management:** Connectivity should be included in the vehicles [225], thus, connected vehicles can increment the positional analysis to supply more information on the environment. Acquisition, storage, and processing of the data of these vehicles and their management are highly challenging.

Alsulami and colleagues [226] addressed the allocation of 5G vehicular network resources for empowering network communication. Moreover, they developed a novel Federated Deep Reinforcement Learning (FDRL) based on the vehicle communication method. They ultimately presented a UAV-aided vehicular communication system based on FDRL-based UAVs for optimizing QoS of 5G and 6G.

An edge server must include enough resources to be able to transfer services from the source to the target nodes. In reality, however, the edge node does not have sufficient resources, which may lead to overload when several requests arrive. Using an integrated fuzzy logic [227], an approach was developed to determine the best resources for vehicles for different conditions. These situations include the quality of the inter-vehicular networks, their size and longevity, the number of accessible resources, and the application requirements. The management strategies were addressed in each edge node (fog node) by channeling resource management in FeRANs [228]. The QoS was enhanced with an emphasis on real-time vehicular services. Two schemes were introduced in this regard: fog resource reservation and fog resource reallocation. This approach managed to increase the on-hop probability for real-time vehicular services even in cases where the fog resource was loaded.

v. **Security and Privacy:** Security and privacy issues are of crucial significance in vehicular crowd sensing, highlighting the need for preserving the users' identity and location. Diverse approaches have been developed concerning crowdsensing and vehicle-based sensing. EC can resolve these issues. Basudan et al. [229] employed EC to introduce a privacy-preserving protocol to alleviate security in a crowdsensing-based road monitoring system. The mentioned approach introduces certificateless aggregate signcryption schemes with a prominent role in decreasing communication overhead and accelerating the verification process. This scheme exhibited minimal computational costs relative to the other schemes. Moreover, the system model considered a control center, RSUs, cloud servers, and vehicles as a part of the road condition monitoring system.

As an inseparable element of intelligent transportation, the vehicular communication network supports diverse mobile applications. Therefore, a secure method should be established to provide effective data sharing. A data sharing scheme was developed by Fan et al. [230], capable of analyzing a multi-authority CPABE scheme through efficient decryption while preserving a CPABE system against collusion attacks. The major decryption is put forward to the cloud. An effective user and multi-authority CPABE was also developed to ensure forward and backward security. Fan and colleagues [230] addressed a novel multiauthority CPABE scheme with the most efficient decryption to realize data access control in a vehicle network and present an efficient user- and attribute-revocation method. Numerous security issues may emerge due to the fast-growing vehicular edge networks. Therefore, more dynamic frameworks are required for better encryption of information to prevent security and privacy violations and reach a more secure EC system.

In VANETs, establishing trust among vehicles is crucial to ensure the secure integrity and reliability of applications. Trusted sources, which provide credible information from nearby vehicles, play a significant role in this regard. Soleymani and coworkers [231] have developed a fuzzy trust model that offers higher speed, accuracy, and reliability. This model incorporates a series of security checks to verify the credibility of the received information. It effectively handles data uncertainty in both line-of-sight and non-line-of-sight scenarios. Additionally, the model is capable of detecting defective nodes and identifying unauthorized attackers, enhancing the overall security of the network. By employing this fuzzy trust model, VANETs can establish a trusted environment where vehicles can rely on the information received from other vehicles, ensuring the integrity and reliability of applications in various driving scenarios.

9. Critical challenges, open issue, and future work

Decentralizing the service infrastructure in VEC brings several advantages, including reduced latency, efficient energy utilization, and increased throughput. To leverage these benefits, a minimum number of vehicles are equipped with sensors for processing and wireless communication, offering potential advantages such as improved safety, efficiency, and convenience. However, there are several challenges that need to be addressed in the VEC. The following sections provide a detailed explanation of each of these issues.

9.1. Critical challenges

1. Mobility:

Traditional sensor network models are designed for static environments, while ad hoc networks primarily consider limited mobility scenarios involving laptops and manual devices. In contrast, vehicular networks are characterized by continuous mobility, which poses unique challenges. The mobility patterns of vehicles exhibit a strong correlation, but each vehicle interacts with a constantly changing set of neighbors. This dynamic nature makes reputation-based approaches less applicable, as vehicles may not have sufficient data to evaluate the trustworthiness of other vehicles. Additionally, the short-lived interactions between vehicles limit the feasibility of protocols that rely on sender-receiver interactions. To address these challenges, an enhanced mobility model is necessary. This model should provide information on vehicular speed, predict vehicular reputation, and capture temporal and spatial distributions. By understanding vehicular attributes and mobility patterns, communication and computational resource utilization can be improved. Unlike conventional data centers, edge devices in VEC are distributed across diverse platforms, introducing heterogeneity that requires optimization of QoS across these platforms. Therefore, an efficient mobility model is crucial for studying mobility patterns in different environments and optimizing resource utilization. This model should take into account the dynamic nature of vehicular networks and provide insights into vehicular attributes and behavior to enhance the overall performance of VEC.

2. Routing and forwarding:

Routing and forwarding in VEC pose several challenges related to the switching of edge servers and their services as vehicles move from one location to another. Two key issues are edge server switching and service switching:

1) **Edge server switching:** Due to the high-speed movement of vehicles, it is challenging to determine which vehicle is receiving services from which base station or edge server based on traffic and public transportation information. Predicting the next move of vehicles remains an open research problem despite the development of various techniques. This issue requires effective methods to track and predict the movements of vehicles to ensure seamless handover of services between edge servers.
2) **Service switching:** When vehicles change their position and move to a different edge server, their associated services need to be transferred to the new server. This requires efficient mechanisms to predict the QoS for service recommendations. Comput and colleagues [232] developed an algorithm to address this problem by predicting the QoS requirements for recommending suitable services to vehicles during the switching process. Despite the efficiency of this algorithm for mobile users, it might not be very efficient in a vehicular environment. Therefore, a timely and reliable service transmission between vehicles and edge servers to maintain the QoS in a vehicular environment is a complex and delicate issue.

3. Content caching:

Content caching (e.g. prefetching and cooperative caching) can also be implemented in VEC. The caching contents may include elements that were not requested by vehicles, but they can take these contents over a wireless connection. It might be useful for the vehicles to save or forward these unrequested contents (e.g., alarms upon trouble). Some gaps have yet remained in caching policies that encompass the most effective temporal and spatial scopes of the vehicular contents, for instance, caching-in contents out of their spatial scope (e.g., emergency signals on the far-off side but still in the relevance area) and also caching-out the old contents (e.g., traffic congestion an hour ago) with the following technical implications:

- V2V communication can increment the network capacity in terms of content caching, however, it fails in validating a reliable and high-rate data service due to the dynamic and uncertain nature of the network and its strict channel conditions.
- As the SRSUs are situated in various places and different network operators can possess them, the cooperation of the SRSUs for content provision must be regarded in terms of the pricing models.
- A caching scheme must be developed to increment content hit rate with the least handover costs, by the identification of cache size splitting, prevalent content updates, and ensuring mobility-aware caching for smooth handover even for high-mobility vehicles.

These vehicular caching systems need such strategies, considering the topographies and network configurations.

4. Deployment of network elements:

An adequate number of network elements can improve the performance of the network on a large scale. Regarding the costly deployment of the network equipment, a proper number of network elements should be optimally installed. The main issue is the proper location to maximize the efficiency of vehicular networks. Moreover, the cost and the position of edge servers and SRSUs must be optimized to make the best use of the available resources. Regarding the variable urban traffic distribution, a higher number of servers must be installed in crowded areas. Given the essential role of servers in the transmission of traffic packets, the SRSUs in the proximity of the servers guide the traffic packets to the infrastructure with no need for multi-hop communications. Using these infrastructures, the packets will be transferred to other nodes within the network. Access to the infrastructure through a smaller number of hops can decrease the receiving time of servers. Thus, an optimal model is required capable of determining the minimum number of needed edge servers and SRSUs to minimize the deployment cost while maximizing the QoS.

5. Task migration:

Vehicle users must outsource computationally demanding and delay-sensitive operations to edge servers owing to the capacity limitations. Optimizing task migration decisions is essential given the dynamic channel environment and frequently changing topology.

6. Resource management:

The computation and storage capabilities of VEC are constrained in comparison to CC. Therefore, it is important to know how to manage these resources. The optimization of resource allocation is a difficult task due to changing resource demands, various application features, and complex traffic situations.

7. Security and Privacy:

The major application in VEC involves offloading the computation-intensive tasks with strict delay to the edge of networks. The offloaded tasks include sensitive and private data. Therefore, data confidentiality should be guaranteed to avoid any type of information leakage. To avoid the modification of forwarded or stored data, the integrity requirements should be fulfilled. Moreover, vehicular users need a verifiable computing scheme to make sure on receiving correct computation results from edge servers.

Security and privacy are two major issues in VEC with a huge impact on the deployment and development of VEC systems. The great mobility of vehicles has hindered solving the problem of trustworthiness among nodes. Regarding the heterogeneity of VEC due to the presence of numerous and various devices in vehicular networks, conventional trust and authentication schemes will fail in this system. The clouds are placed near users which makes the edge servers prone to attacks due to their public deployment with no physical isolation. Moreover, they may be turned malicious due to the user's curiosity. Studies on the security and privacy of VEC are in their infancy as a lot of problems have remained unsolved. This opens fascinating research opportunities which explain restless efforts in this field.

These challenges highlight the need for robust solutions that can accurately predict vehicle movements and provide seamless service switching between edge servers. By addressing these issues, VEC can enhance the overall performance and QoS for vehicles in dynamic and fast-paced environments.

9.2. Open issue

1. QoS:

Vehicle networks have a wide range of applications, which are primarily divided into safety applications and non-safety applications. Various types of applications are anticipated to have different QoS requirements. While some non-safety applications, such as multimedia downloading, can tolerate some delays, safety applications, which include collision avoidance and traffic control, have severe delay requirements that should be handled as quickly as possible. Therefore, a worry in VEC is offering a flexible scheduling strategy to ensure the QoS of various applications according to their priority.

2. Scalability:

Applications are coming in in abundance and growing quickly. As a result, there are increased demands for low latency, excellent reliability, and abundant computing and storage capacity. For completing various task kinds, it is essential to schedule resources effectively and perform connectivity management effectively. Furthermore, unlike traditional clouds, vehicle users in VEC may be dispersed unevenly throughout vehicular networks.

Vehicle densities are changing over time in various locations. The system that was created should be able to adjust to changing network conditions.

3. Monetary advantage:

Resource sharing is at the heart of the VEC implementation.

If they can be fairly compensated, resource owners are prepared to share their resources. The pricing system assumes a crucial role in this scenario. How can the values of various resources be quantified to balance the profits of both resource users and resource providers The distribution of retained earnings among connected companies, such as cloud service providers, mobile service providers, and edge service providers, should be taken into consideration by resource providers.

4. Big data analytics:

It is essential to efficiently classify and evaluate vehicle data because of the variances, temporal and spatial data elements, large volumes, and various data elements. BD integration with SDN and NFV in the real-world setting of vehicle offloading is still a controversial topic [10].

5. Cooperative vehicular environment:

A significant method to manage severe channel fading and offload data from infrastructures is cooperative communication. In a scenario involving cooperative computing, several vehicles can participate by acting as cooperators to assist their nearby counterparts in carrying out computing activities and delivering the outcome to the beneficiary. In particular, the cooperation strategy should be developed by jointly considering user behaviors and vehicle status. Additionally, it is crucial to combine promising optimization theory with BD.

Users in vehicle networks could share interests in particular content due to comparable social activities. Contrarily, content is transmitted by RSU-based MEC because direct connection between vehicles is ineffective. Consequently, researching different V2V and V2I communication protocols is essential for enhancing performance, particularly in the case of V2V and V2I offloading [208].

6. Data timeliness:

The high velocity, dynamic temporal/spatial fluctuation, and rigorous timeliness requirements of BD gathered from the transportation, ICT, energy, and social sectors could cause the data to instantly become outdated. More consideration should be given to data timeliness. To ensure that the data of end-user devices, surroundings, and systems can be obtained in a timely manner and that successful strategies for energy-efficient VEC can be realized, it is imperative to create efficient data collecting and processing approaches [10].

9.3. Future work

1. Energy Efficiency and Several Applications for Vehicles:

The subject of offloading computation to independent computing tasks has been examined in numerous research that has already been published. However, implementing energy-efficient ComOf decisions continues to be a difficult topic because of the inter-task dependency in multiple devices that frequently occurs in IoV systems.

A crucial EC strategy for effectively enhancing the processing capacity of IoT sensors is ComOf. Additionally, computation offloading can reduce calculation energy consumption at the expense of increased transmission energy use. As a result, one of the main difficulties in the ComOf problem is balancing the trade-off between computation and communication costs to maximize offloading solutions.

2. Interference Coordination:

In VEC, interference is a critical problem, particularly when it comes to co-channel transmission when caching and offloading are occurring. D2D is one of the strategies used by VEC to meet the demands of 5G. It is necessary to investigate how to handle the conflict between D2D and conventional communications.

3. Privacy and Security:

Proper privacy data security methods should be implemented since the majority of proposed VEC frameworks incorporated a variety of information, ranging from mobile devices to infrastructures. To ensure a secure connection and uphold data confidentiality during the offloading procedures, the underlying threat must be investigated. Since the end-user must obtain trustworthy data for safer driving objectives, false information conveyance and traffic scene forgeries are among the security issues in VEC. To prevent data theft, it is also important to secure vehicle content.

4. Improved Vehicle Communications:

When you're operating a vehicle, every second counts. The information gathered by local sensors in vehicles would then be continuously uploaded to the closest edge device. To prevent service interruptions and QoS loss, energy and power consumption at the edge should be taken into account.

Additionally, certain situations, such as severe traffic congestion, unforeseen weather conditions, or unforeseen road construction activities, require significant QoS enhancement to handle periodic high traffic loads. As a result, more research is required to improve and control QoS in the context of V2X while considering a heterogeneous edge-based environment.

5. Offloading AI Algorithms:

Cutting-edge advances in AI techniques have opened up a slew of new opportunities for ITS, aided in particular by the vehicles' intelligent sensors, which are constantly improving over time and allowing the vehicles to better assess their surroundings [233]. Both the paradigms of task offloading, and vehicle data have seen the use of AI or ML algorithms.

In most cases, ML algorithms demand higher computational resources, which can be provided by edge servers, such as VEC or VCC.

In terms of Al algorithms, further study in this field is urgently needed, given the dynamic nature of the vehicles environment, the variety of application needs, and the strict delay requirements of vehicular applications and services [234].

In most cases, ML algorithms demand higher computational resources, which can be provided by edge servers, such as VEC or VCC. Numerous vehicular-focused AI applications implementing DL frameworks also require offloading for processing, in addition to AI algorithms for offloading to make optimal decisions.

10. Conclusion

In conclusion, this study delved into the transformative landscape of EC and its pivotal role in reshaping the future of computational paradigms. With its attributes of reduced latency, optimized bandwidth usage, and enhanced data privacy, EC sets the stage for critical advancements across various application domains, ranging from autonomous vehicles to smart homes.

The focal point of our investigation was the intricate domain of computation offloading within the context of Edge-IoV networks, particularly in the industry 4.0 era and beyond. Navigating through the collaborative environment of cloud-edge computing, our research addresses the complex decision-making processes involved in computation offloading, necessitating resource management and allocation across multiple entities.

Our exploration extended to the promising architecture of VEC, designed to bolster scalability, facilitate real-time data delivery, and enhance mobility within vehicular networks. The impact of VEC on computational performance, promoting intelligent vehicular computing and optimal utilization of underutilized computational resources within vehicles, was a key focal point.

This study also conducted an exhaustive review of existing works in the realm of VEC, shedding light on the current state of research and laying the foundation for future advancements. To guide further exploration, we have proposed several avenues for future research and identified open challenges that beckon the attention of academics and researchers in this dynamic field.

In reflection, our discussions aim to serve not only the research community but also offer valuable insights for industrial practitioners. The comprehension of computation offloading within the EC framework presented in this article can inform the development of superior systems, incorporating advanced resource management and computing placement mechanisms.

Despite these contributions, we acknowledge the need for clarity in our conclusion. To better serve our readers, we commit to refining and elucidating our concluding remarks, ensuring a more seamless understanding of the implications and findings of our study.

11. Intellectual property

we declare that we have no patents, copyrights, or any other intellectual property that may pose a conflict of interest in relation to this research.

Funding sources

This research has been funded by [in part by the Projects **BIDANET:** Parametric Big Data Analytics over Wireless Networks (Vote No: 9696300), and in part by the **EMOSEN**-Energy Efficient MIMO-Based Wireless Transmission for SWIPT-Enabled Network (UPM/800–3/3/1/GPB/2019/9671600, Vote No: 9671600) and in part by the **IGNITE** – Interference Modelling for 5G and FSS Coexistence at mmWave with Climate Change Considerations in the Tropical Region (FRGS/1/2021/TK0/UPM/01/1, File No.: 03–01-21-2375FR. Vote No: 5540509)], and we affirm that these funding sources had no influence over the design, execution, or reporting of the research.

CRediT authorship contribution statement

Marieh Talebkhah: Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Supervision, Validation, Writing – original draft, Writing – review & editing. **Aduwati Sali:** Formal analysis, Funding acquisition, Project administration, Supervision, Writing – review & editing, Conceptualization, Investigation, Methodology. **Vahid Khodamoradi:** Conceptualization, Writing – review & editing. **Touraj Khodadadi:** Formal analysis, Funding acquisition, Writing – review & editing. **Meisam Gordan:** Conceptualization, Formal analysis, Writing – original draft, Writing – review & editing, Data curation, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] N. Shirzad-Ghaleroudkhani, Q. Mei, M. Gül, A crowdsensing-based platform for transportation infrastructure monitoring and management in smart cities, in, Rise Smart Cities, Butterworth-Heinemann (2022) 609–624, https://doi.org/10.1016/B978-0-12-817784-6.00005-9.

[2] M. Gordan, S.-R. Sabbagh-Yazdi, K. Ghaedi, D.P. Thambiratnam, Z. Ismail, Introduction to monitoring of bridge infrastructure using soft computing techniques, Appl. Methods Bridg. Des. Optim. Pract. 2022 (2022), https://doi.org/10.5772/intechopen.104905.

[3] L. Lin, Data Science Application in Intelligent Transportation Systems: An Integrative Approach for Border Delay Prediction and Traffic Accident Analysis, University at Buffalo, 2015.

[4] S. Sharma, B. Kaushik, A survey on internet of vehicles: applications, security issues & solutions, Veh. Commun. 20 (2019) 100182, https://doi.org/10.1016/j.vehcom.2019.100182.
[5] S.Y. Lien, S.C. Hung, D.J. Deng, C.L. Lai, H.L. Tsai, Low latency radio access in 3GPP local area data networks for V2X: Stochastic optimization and learning, IEEE Internet Things J. 6 (2018) 4867–4879, https://doi.org/10.1109/JIOT.2018.2874883.
[6] C. Chen, L. Liu, T. Qiu, K. Yang, F. Gong, H. Song, ASGR: An artificial spider-web-based geographic routing in heterogeneous vehicular networks, IEEE Trans. Intell. Transp. Syst. 20 (2018) 1604–1620, https://doi.org/10.1109/TITS.2018.2828025.
[7] O. Kaiwartya, A.H. Abdullah, Y. Cao, A. Altameem, M. Prasad, C.T. Lin, X. Liu, Internet of vehicles: motivation, layered architecture, network model, challenges, and future aspects, IEEE Access 4 (2016) 5356–5373, https://doi.org/10.1109/ACCESS.2016.2603219.
[8] C. Chen, J. Hu, T. Qiu, M. Atiquzzaman, Z. Ren, CVCG: cooperative V2V-aided transmission scheme based on coalitional game for popular content distribution in vehicular Ad-Hoc networks, IEEE Trans. Mob. Comput. 18 (2019) 2811–2828, https://doi.org/10.1109/TMC.2018.2883312.
[9] R.A. Dziyauddin, D. Niyato, N.C. Luong, A.A.A. Mohd Atan, M.A. Mohd Izhar, M. H. Azmi, S. Mohd Daud, Computation offloading and content caching and delivery in vehicular edge network: a survey, Comput. Networks 197 (2021) 108228, https://doi.org/10.1016/j.comnet.2021.108228.
[10] Z. Zhou, H. Yu, C. Xu, Z. Chang, S. Mumtaz, J. Rodriguez, BEGIN: big data enabled energy-efficient vehicular edge computing, IEEE Commun. Mag. 56 (2018) 82–89, https://doi.org/10.1109/MCOM.2018.1700910.
[11] H.T. Dinh, C. Lee, D. Niyato, P. Wang, A survey of mobile cloud computing: architecture, applications, and approaches, Wirel. Commun. Mob. Comput. 13 (2013) 1587–1611, https://doi.org/10.1002/wcm.
[12] C.C. Lin, D.J. Deng, C.C. Yao, Resource allocation in vehicular cloud computing systems with heterogeneous vehicles and roadside units, IEEE Internet Things J. 5 (2017) 3692–3700, https://doi.org/10.1109/JIOT.2017.2690961.
[13] N. Fernando, S.W. Loke, W. Rahayu, Mobile cloud computing: a survey, Futur. Gener. Comput. Syst. 29 (2013) 84–106, https://doi.org/10.1016/j.future.2012.05.023.
[14] A. ur Rehman Khan, M. Othman, S.A. Madani, S. Ullah Khan, A survey of mobile cloud computing application models, Int. J. Res. Appl. Sci. Eng. Technol. 7 (2019) 2438–2441. 10.22214/ijraset.2019.6411.
[15] K. Akherfi, M. Gerndt, H. Harroud, Mobile cloud computing for computation offloading: Issues and challenges, Appl. Comput. Informatics 14 (2018) 1–16, https://doi.org/10.1016/j.aci.2016.11.002.
[16] S. Shamshirband, M. Fathi, A.T. Chronopoulos, A. Montieri, F. Palumbo, A. Pescap, Computational intelligence intrusion detection techniques in mobile cloud computing environments: review, taxonomy, and open research issues, J. Inf. Secur. Appl. 55 (2020) 102582, https://doi.org/10.1016/j.jisa.2020.102582.
[17] Y.W. Lin, J.M. Shen, H.C. Weng, Cloud-supported seamless internet access in intelligent transportation systems, Wirel. Pers. Commun. 72 (2013) 2081–2106, https://doi.org/10.1007/s11277-013-1137-5.
[18] S. Bitam, A. Mellouk, S. Zeadally, Vanet-cloud: a generic cloud computing model for vehicular ad hoc networks, IEEE Wirel. Commun. 22 (2015) 96–102, https://doi.org/10.1109/MWC.2015.7054724.
[19] K. Mershad, H. Artail, Finding a STAR in a vehicular cloud, IEEE Intell. Transp. Syst. Mag. 5 (2013) 55–68, https://doi.org/10.1109/MITS.2013.2240041.
[20] S.B. Akintoye, A. Bagula, Improving quality-of-service in cloud/fog computing through efficient resource allocation, Sensors 19 (2019) 1267, https://doi.org/10.3390/s19061267.
[21] M. Talebkhah, A. Sali, M. Marjani, M. Gordan, S.J. Hashim, F.Z. Rokhani, Edge computing : architecture , applications and future perspectives, in: 2020 IEEE 2nd Int. Conf. Artif. Intell. Eng. Technol., IEEE, 2020: pp. 1–6.
[22] Y. Dai, D. Xu, S. Maharjan, Y. Zhang, Joint computation offloading and user association in multi-task mobile edge computing, IEEE Trans. Veh. Technol. 67 (2018) 12313–12325, https://doi.org/10.1109/HPCS.2018.00157.
[23] L.U. Khan, I. Yaqoob, N.H. Tran, S.M.A. Kazmi, T.N. Dang, C.S. Hong, Edge-computing-enabled smart cities: a comprehensive survey, IEEE Internet Things J. 7 (2020) 10200–10232, https://doi.org/10.1109/JIOT.2020.2987070.
[24] P. Mach, Z. Becvar, Mobile edge computing: a survey on architecture and computation offloading, IEEE Commun. Surv. & Tutorials 19 (2017) 1628–1656, https://doi.org/10.1109/COMST.2017.2682318.
[25] N. Abbas, Y. Zhang, A. Taherkordi, T. Skeie, Mobile edge computing: a survey, IEEE Internet Things J. 5 (2018) 450–465, https://doi.org/10.1109/JIOT.2017.2750180.
[26] Y. Mao, C. You, J. Zhang, K. Huang, K.B. Letaief, A survey on mobile edge computing: the communication perspective, IEEE Commun. Surv. Tutorials 19 (2017) 2322–2358, https://doi.org/10.1109/COMST.2017.2745201.
[27] R. Roman, J. Lopez, M. Mambo, Mobile edge computing, Fog et al.: a survey and analysis of security threats and challenges, Futur. Gener. Comput. Syst. 78 (2018) 680–698, https://doi.org/10.1016/j.future.2016.11.009.
[28] Z. Zhou, H. Liao, B. Gu, S. Mumtaz, J. Rodriguez, Resource sharing and task offloading in IoT fog computing: a contract-learning approach, IEEE Trans. Emerg. Top. Comput. Intell. 4 (2020) 227–240, https://doi.org/10.1109/TETCI.2019.2902869.
[29] M. Talebkhah, A. Sali, M. Marjani, M. Gordan, S.J. Hashim, F.Z. Rokhani, IoT and big data applications in smart cities: recent advances, challenges, and critical issues, IEEE Access 9 (2021) 55465–55484, https://doi.org/10.1109/ACCESS.2021.3070905.
[30] L. Yang, H. Yao, J. Wang, C. Jiang, A. Benslimane, Y. Liu, Multi-UAV-enabled load-balance mobile-edge computing for IoT networks, IEEE Internet Things J. 7 (2020) 6898–6908, https://doi.org/10.1109/JIOT.2020.2971645.
[31] M. Gordan, Z. Ismail, K. Ghaedi, Z. Ibrahim, H. Hashim, H. Ghayeb, M. Talebkhah, A brief overview and future perspective of unmanned aerial systems for in-service structural health monitoring, Eng. Adv. 1 (2021) 9–15. 10.26855/ea.2021.06.002.
[32] K. Hayawi, Z. Anwar, A.W. Malik, Z. Trabelsi, Airborne computing: a toolkit for uav-assisted federated computing for sustainable smart cities, IEEE Internet Things J. (2023), https://doi.org/10.1109/JIOT.2023.3292308.
[33] A. Gupta, S.K. Gupta, A survey on green unmanned aerial vehicles-based fog computing: challenges and future perspective Akshita, Trans. Emerg. Telecommun. Technol. 33 (2022) e4603.
[34] S. Das, R.S. Rao, I. Das, V. Jain, N. Singh, Cloud computing enabled big-data analytics in wireless ad-hoc networks, CRC Press, 2022.
[35] C. Zhu, J. Tao, G. Pastor, Y. Xiao, Y. Ji, Q. Zhou, Y. Li, A. Yla-jaaski, Folo: latency and quality optimized task allocation in vehicular fog computing, IEEE Internet Things J. 6 (2018) 4150–4161, https://doi.org/10.1109/JIOT.2018.2875520.
[36] C. Zhu, Y.-H. Chiang, A. Mehrabi, Y. Xiao, Chameleon : Latency and resolution aware task offloading for visual-based assisted driving, IEEE Trans. Veh. Technol. 68 (2019) 9038–9048, https://doi.org/10.1109/TVT.2019.2924911.
[37] X.Q. Pham, T.D. Nguyen, V.D. Nguyen, E.N. Huh, Joint node selection and resource allocation for task offloading in scalable vehicle-assisted multi-access edge computing, Symmetry (Basel). 11 (2019) 58, https://doi.org/10.3390/sym11010058.
[38] H. Liao, Y. Mu, Z. Zhou, M. Sun, Z. Wang, C. Pan, Blockchain and learning-based secure and intelligent task offloading for vehicular, IEEE Trans. Intell. Transp. Syst. 22 (2020) 4051–4063.
[39] L. Yang, H. Zhang, M. Li, J. Guo, H. Ji, Mobile edge computing empowered energy efficient task offloading in 5G, IEEE Trans. Veh. Technol. 67 (2018) 6398–6409, https://doi.org/10.1109/TVT.2018.2799620.
[40] D. Xu, Y. Li, X. Chen, J. Li, P. Hui, S. Chen, J. Crowcroft, A survey of opportunistic offloading, IEEE Commun. Surv. Tutorials 20 (2018) 2198–2236.
[41] H. Zhou, H. Wang, X. Li, V.C.M. Leung, A survey on mobile data offloading technologies, IEEE Access 6 (2018) 5101–5111, https://doi.org/10.1109/ACCESS.2018.2799546.
[42] J. Moura, D. Hutchison, Game theory for multi-access edge computing: survey, use cases, and future trends, IEEE Commun. Surv. Tutorials 21 (2018) 260–288.
[43] L. Mendiboure, M.A. Chalouf, F. Krief, Edge computing based applications in vehicular environments: comparative study and main issues, J. Comput. Sci. Technol. 34 (2019) 869–886, https://doi.org/10.1007/s11390-019-1947-3.
[44] Z. Zhou, H. Liao, X. Zhao, B. Ai, M. Guizani, Reliable task offloading for vehicular fog computing under information asymmetry and information uncertainty, IEEE Trans. Veh. Technol. 68 (2019) 8322–8335, https://doi.org/10.1109/TVT.2019.2926732.
[45] T. Wang, P. Li, X. Wang, Y. Wang, T. Guo, T. Wang, A comprehensive survey on mobile data offloading in heterogeneous network, Wirel Networks 25 (2019) 573–584, https://doi.org/10.1007/s11276-017-1576-0.
[46] H. Zhou, H. Wang, X. Chen, X. Li, S. Xu, Data offloading techniques through vehicular ad hoc networks : a survey, IEEE Access 6 (2018) 65250–65259, https://doi.org/10.1109/ACCESS.2018.2878552.
[47] D.J. Deng, S.Y. Lien, C.C. Lin, S.C. Hung, W.B. Chen, Latency control in software-defined mobile-edge vehicular networking, IEEE Commun. Mag. 55 (2017) 87–93, https://doi.org/10.1109/MCOM.2017.1601165.
[48] X. Xu, R. Gu, F. Dai, L. Qi, S. Wan, Multi-objective computation offloading for Internet of Vehicles in cloud-edge computing, Wirel. Networks (2019) 1–19, https://doi.org/10.1007/s11276-019-02127-y.
[49] S. Wan, Y. Zhang, J. Chen, On the construction of data aggregation tree with maximizing lifetime in large-scale wireless sensor networks, IEEE Sens. J. 16 (2016) 7433–7440, https://doi.org/10.1109/JSEN.2016.2581491.
[50] L. Qi, R. Wang, C. Hu, S. Li, Q. He, X. Xu, Time-aware distributed service recommendation with privacy-preservation, Inf. Sci. (Ny) 480 (2019) 354–364, https://doi.org/10.1016/j.ins.2018.11.030.
[51] S. Choo, J. Kim, S. Pack, Optimal task offloading and resource allocation in software-defined vehicular edge computing, in: 9th Int. Conf. Inf. Commun. Technol. Converg. ICT Converg. Powered by Smart Intell. (ICTC), IEEE, 2018: pp. 251–256. 10.1109/ICTC.2018.8539726.
[52] A. Ashok, P. Steenkiste, F. Bai, Vehicular cloud computing through dynamic computation offloading, Comput. Commun. 120 (2018) 125–137, https://doi.org/10.1016/j.comcom.2017.12.011.
[53] S. Yuan, Y. Fan, Y. Cai, A survey on computation offloading for vehicular edge computing, in: 2019 7th Int. Conf. Inf. Technol. IoT Smart City, 2019: pp. 107–112. 10.1145/3377170.3377228.
[54] X. Cheng, X. Zhou, C. Jiang, J. Wan, Towards computation offloading in edge computing: A survey, in: High-Performance Comput. Appl. Numer. Simul. Edge Comput. ACM ICS 2018 Int. Work, HPCMS HiDEC, Springer, Singapore, 2019, pp. 3–15, https://doi.org/10.1007/978-981-32-9987-0_1.
[55] A.B. De Souza, P.A.L. Rego, T. Carneiro, J.D.C. Rodrigues, P.P.R. Filho, J.N. de Souza, V. Chamola, V.H.C. de Albuquerque, B. Sikdar, Computation offloading for vehicular environments: a survey, IEEE Access 8 (2020) 198214–198243, https://doi.org/10.1109/ACCESS.2020.3033828.
[56] L. Liu, C. Chen, Q. Pei, S. Maharjan, Y. Zhang, Vehicular edge computing and networking: a survey, Mob. Networks Appl. 26 (2021) 1145–1168, https://doi.org/10.1007/s11036-020-01624-1.

[57] M. Wang, H. Yi, F. Jiang, L. Lin, M. Gao, Review on offloading of vehicle edge computing, J. Artif. Intell. Technol. (2022), https://doi.org/10.37965/jait.2022.0120.
[58] A.K. Naren, N. Gaurav, A.P. Sahu, G.S.S. Dash, V. Chalapathi, A survey on computation resource allocation in IoT enabled vehicular edge computing, Complex Intell. Syst. 8 (2022) 3683–3705, https://doi.org/10.1007/s40747-021-00483-x.
[59] A. Waheed, M.A. Shah, S.M. Mohsin, A. Khan, C. Maple, S. Aslam, S. Shamshirband, A comprehensive review of computing paradigms, enabling computation offloading and task execution in vehicular networks, IEEE Access 10 (2022) 3580–3600, https://doi.org/10.1109/ACCESS.2021.3138219.
[60] B. Kar, W. Yahya, Y.D. Lin, A. Ali, Offloading using traditional optimization and machine learning in federated cloud-edge-fog systems: a survey, IEEE Commun. Surv. Tutorials (2023), https://doi.org/10.1109/COMST.2023.3239579.
[61] J. Xie, Y. Jia, Z. Chen, Z. Nan, L. Liang, Efficient task completion for parallel offloading in vehicular fog computing, China Commun. 16 (2019) 42–55. 10.23919/JCC.2019.11.004.
[62] O.P. Mahela, N. Gupta, M. Khosravy, N. Patel, Comprehensive overview of low voltage ride through methods of grid integrated wind generator, IEEE Access 7 (2019) 99299–99326, https://doi.org/10.1109/ACCESS.2019.2930413.
[63] Z. Zhou, P. Liu, J. Feng, Y. Zhang, S. Mumtaz, J. Rodriguez, Computation resource allocation and task assignment optimization in vehicular fog computing: a contract-matching approach, IEEE Trans. Veh. Technol. 68 (2019) 3113–3125, https://doi.org/10.1109/TVT.2019.2894851.
[64] Q. Wu, H. Ge, H. Liu, Q. Fan, Z. Li, Z. Wang, A task offloading scheme in vehicular fog and cloud computing system, IEEE Access 8 (2019) 1173–1184, https://doi.org/10.1109/ACCESS.2019.2961802.
[65] M. Muzakkir Hussain, M.M. Sufyan Beg, Fog computing for internet of things (IoT)-aided smart grid architectures, Big Data Cogn. Comput. 3 (2019) 8, https://doi.org/10.3390/bdcc3010008.
[66] T.S.J. Darwish, K.A. Bakar, Fog based intelligent transportation big data analytics in the internet of vehicles environment: motivations, architecture, challenges, and critical issues, IEEE Access 6 (2018) 15679–15701, https://doi.org/10.1109/ACCESS.2018.2815989.
[67] Z. Ning, K. Zhang, X. Wang, L. Guo, X. Hu, J. Huang, B. Hu, R.Y.K. Kwok, Intelligent edge computing in internet of vehicles: a joint computation offloading and caching solution, IEEE Trans. Intell. Transp. Syst. 22 (2020) 2212–2225.
[68] P. Wang, C. Yao, Z. Zheng, G. Sun, L. Song, Joint task assignment transmission and computing resource allocation in multi-layer mobile edge computing systems, IEEE Internet Things J. 6 (2018) 2872–2884, https://doi.org/10.1109/JIOT.2018.2876198.
[69] Q. Zhang, L. Gui, F. Hou, J. Chen, S. Zhu, F. Tian, Dynamic task offloading and resource allocation for mobile edge computing in dense cloud RAN, IEEE Internet Things J. 7 (2020) 3282–3299, https://doi.org/10.1109/JIOT.2020.2967502.
[70] Y. Chen, W. Zheng, W. Li, Y. Huang, Large group activity security risk assessment and risk early warning based on random forest algorithm, Pattern Recognit. Lett. 144 (2021) 1–5, https://doi.org/10.1016/j.patrec.2021.01.008.
[71] J. Zhao, J. Liu, J. Jiang, F. Gao, Efficient deployment with geometric analysis for mmWave UAV communications, IEEE Wirel. Commun. Lett. 9 (2020) 1115–1119, https://doi.org/10.1109/LWC.2020.2982637.
[72] Z. Ning, J. Huang, X. Wang, J.J.P.C. Rodrigues, L. Guo, Mobile edge computing-enabled internet of vehicles: toward energy-efficient scheduling, IEEE Netw. 33 (2019) 198–205, https://doi.org/10.1109/MNET.2019.1800309.
[73] N. Kumar, S. Zeadally, J.J.P.C. Rodrigues, Vehicular delay-tolerant networks for smart grid data management using mobile edge computing, IEEE Commun. Mag. 54 (2016) 60–66, https://doi.org/10.1109/MCOM.2016.7588230.
[74] C. Yu, B. Lin, P. Guo, W. Zhang, S. Li, R. He, Deployment and dimensioning of fog computing-based internet of vehicle infrastructure for autonomous driving, IEEE Internet Things J. 6 (2018) 149–160, https://doi.org/10.1109/JIOT.2018.2875750.
[75] K. Zhang, J. Zhang, X. Ma, C. Yao, L. Zhang, Y. Yang, J. Wang, J. Yao, H. Zhao, History matching of naturally fractured reservoirs using a deep sparse autoencoder, SPE J. 26 (2021) 1700–1721, https://doi.org/10.2118/205340-PA.
[76] X. Ma, K. Zhang, L. Zhang, C. Yao, J. Yao, H. Wang, W. Jian, Y. Yan, Data-driven niching differential evolution with adaptive parameters control for history matching and uncertainty quantification, SPE J. 26 (2021) 993–1010, https://doi.org/10.2118/205014-PA.
[77] Z. Xiong, N. Xiao, F. Xu, X. Zhang, Q. Xu, K. Zhang, C. Ye, An equivalent exchange based data forwarding incentive scheme for socially aware networks, J. Signal Process. Syst. 93 (2021) 249–263, https://doi.org/10.1007/s11265-020-01610-6.
[78] Y. Dai, D. Xu, S. Maharjan, Y. Zhang, Joint load balancing and offloading in vehicular edge computing and networks, IEEE Internet Things J. 6 (2018) 4377–4387, https://doi.org/10.1109/JIOT.2018.2876298.
[79] Y. Jie, C. Guo, K.R. Choo, C.Z. Liu, M. Li, Game-theoretic resource allocation for fog- based industrial internet of things environment, IEEE Internet Things J. 7 (2020) 3041–3052, https://doi.org/10.1109/JIOT.2020.2964590.
[80] A. Mebrek, A. Yassine, Intelligent resource allocation and task offloading model for IoT applications in fog networks: a game-theoretic approach, IEEE Trans. Emerg. Top. Comput. Intell. (2021), https://doi.org/10.1109/TETCI.2021.3102214.
[81] M. Liwang, J. Wang, Z. Gao, X. Du, M. Guizani, Game theory based opportunistic computation offloading in cloud-enabled IoV, IEEE Access 7 (2019) 32551–32561, https://doi.org/10.1109/ACCESS.2019.2897617.
[82] C. Swain, M.N. Sahoo, A. Satpathy, K. Muhammad, S. Bakshi, J.J.P.C. Rodrigues, V.H.C. De Albuquerque, METO: matching-theory-based efficient task offloading in IoT-fog interconnection networks, IEEE Internet Things J. 8 (2020) 12705–12715, https://doi.org/10.1109/JIOT.2020.3025631.
[83] F. Chiti, R. Fantacci, B. Picano, A matching theory framework for tasks offloading in fog computing for IoT systems, IEEE Internet Things J. 5 (2018) 5089–5096, https://doi.org/10.1109/JIOT.2018.2871251.
[84] A. Abouaomar, A. Kobbane, S. Cherkaoui, Matching-game for user-fog assignment, in: 2018 IEEE Glob. Commun. Conf., IEEE, 2018: pp. 1–6.
[85] C. Swain, M.N. Sahoo, A. Satpathy, K. Muhammad, S. Bakshi, J.J.P.C. Rodrigues, A-DAFTO : artificial cap deferred acceptance-based fair task offloading in complex IoT-fog networks, IEEE Trans. Consum. Electron. 69 (2023) 914–926, https://doi.org/10.1109/TCE.2023.3262995.
[86] H. Guo, J. Liu, J. Zhang, W. Sun, N. Kato, Mobile-edge computation offloading for ultradense IoT networks, IEEE Internet Things J. 5 (2018) 4977–4988, https://doi.org/10.1109/JIOT.2018.2838584.
[87] A.A. Ateya, A. Muthanna, R. Kirichek, M. Hammoudeh, A. Koucheryavy, Energy- and latency-aware hybrid offloading algorithm for UAVs, IEEE Access 7 (2019) 37587–37600.
[88] M. Messous, S. Senouci, H. Sedjelmaci, S. Cherkaoui, A game theory based efficient computation offloading in an UAV network, IEEE Trans. Veh. Technol. 68 (2019) 4964–4974.
[89] H. Guo, J. Liu, UAV-enhanced intelligent offloading for internet of things at the edge, IEEE Trans. Ind. Informatics 16 (2019) 2737–2746, https://doi.org/10.1109/TII.2019.2954944.
[90] Z. Yu, Y. Gong, S. Gong, Y. Guo, Joint task offloading and resource allocation in UAV-enabled mobile edge computing, IEEE Internet Things J. 7 (2020) 3147–3159, https://doi.org/10.1109/JIOT.2020.2965898.
[91] J. Contreras-Castillo, S. Zeadally, J.A. Guerrero-Ibanez, Internet of vehicles: architecture protocols, and security, IEEE Internet Things J. 5 (2017) 3701–3709, https://doi.org/10.1109/JIOT.2017.2690902.
[92] C. Long, Y. Cao, T. Jiang, Q. Zhang, Edge computing framework for cooperative video processing in multimedia IoT systems, IEEE Trans. Multimed. 20 (2018) 1126–1139, https://doi.org/10.1109/TMM.2017.2764330.
[93] M. Babar, F. Arif, M.A. Jan, Z. Tan, F. Khan, Urban data management system: Towards Big Data analytics for Internet of Things based smart urban environment using customized Hadoop, Futur. Gener. Comput. Syst. 96 (2019) 398–409, https://doi.org/10.1016/j.future.2019.02.035.
[94] Y. Zhang, C. Wang, H. Wei, Parking reservation auction for parked vehicle assistance in vehicular fog computing, IEEE Trans. Veh. Technol. 68 (2019) 3126–3139.
[95] A. Lohachab, Bootstrapping urban planning : addressing big data issues in smart cities, in: Res. Anthol. Big Data Anal. Archit. Appl., IGI Global, 2022: pp. 1329–1358.
[96] R. Chaudhary, N. Kumar, S. Zeadally, Network service chaining in fog and cloud computing for the 5G environment: data management and security challenges, IEEE Commun. Mag. 55 (2017) 114–122, https://doi.org/10.1109/MCOM.2017.1700102.
[97] Q.V. Pham, F. Fang, V.N. Ha, M.J. Piran, M. Le, L.B. Le, W.J. Hwang, Z. Ding, A Survey of multi-access edge computing in 5G and beyond: fundamentals, technology integration, and state-of-the-art, IEEE Access 8 (2020) 116974–117017, https://doi.org/10.1109/ACCESS.2020.3001277.
[98] R. Basir, S. Qaisar, M. Ali, M. Aldwairi, M.I. Ashraf, A. Mahmood, M. Gidlund, Fog computing enabling industrial internet of things: state-of-the-art and research challenges, Sensors (Switzerland) 19 (2019) 4807, https://doi.org/10.3390/s19214807.
[99] M. Talebkhah, A. Sali, M. Gordan, S.J. Hashim, F.Z. Rokhani, Comprehensive review on development of smart cities using industry 4 . 0 technologies, IEEE Access (2023).
[100] M.R. Anawar, S. Wang, M.A. Zia, A.K. Jadoon, U. Akram, S. Raza, Fog computing: an overview of big IoT data analytics, Wirel. Commun. Mob. Comput. 2018 (2018).
[101] A. Molinaro, C. Campolo, 5G for V2X communications, 5G Italy White Eb. (2019).
[102] T. Alladi, V. Kohli, V. Chamola, F.R. Yu, Securing the internet of vehicles: a deep learning-based classification framework, IEEE Netw. Lett. 3 (2021) 94–97, https://doi.org/10.1109/lnet.2021.3058292.
[103] D.S. Pereira, M.R. De Morais, L.B.P. Nascimento, P.J. Alsina, V.G. Santos, D.H. S. Fernandes, M.R. Silva, Zigbee protocol-based communication network for multi-unmanned aerial vehicle networks, IEEE Access 8 (2020) 57762–57771, https://doi.org/10.1109/ACCESS.2020.2982402.
[104] H. Zhou, W. Xu, J. Chen, W. Wang, Evolutionary V2X technologies toward the internet of vehicles: challenges and opportunities, Proc. IEEE 108 (2020) 308–323, https://doi.org/10.1109/JPROC.2019.2961937.
[105] R. Yadav, W. Zhang, S. Member, I.A. Elgendy, G. Dong, M. Shafiq, S. Member, A. A. Laghari, S. Prakash, Smart healthcare: RL-based task offloading scheme for edge-enable sensor networks, IEEE Sens. J. 21 (2021) 24910–24918.
[106] P. Zhang, M. Zhou, G. Fortino, Security and trust issues in Fog computing: a survey, Futur. Gener. Comput. Syst. 88 (2018) 16–27, https://doi.org/10.1016/j.future.2018.05.008.
[107] Z. Wan, D. Xu, D. Xu, I. Ahmad, Joint computation offloading and resource allocation for NOMA-based multi-access mobile edge computing systems, Comput. Networks 196 (2021) 108256, https://doi.org/10.1016/j.comnet.2021.108256.
[108] S. Chen, Q. Li, M. Zhou, Recent advances in collaborative scheduling of computing tasks in an edge computing paradigm, Sensors 21 (2021) 779.
[109] X. Zhang, J. Zhang, Z. Liu, Q. Cui, X. Tao, S. Wang, MDP-based task offloading for vehicular edge computing under certain and uncertain transition, IEEE Trans.

Veh. Technol. 69 (2020) 3296–3309, https://doi.org/10.1109/TVT.2020.2965159.
[110] X. Shen, Z. Chang, X. Xie, S. Niu, Task offloading strategy of vehicular networks based on improved bald eagle search optimization algorithm, Appl. Sci. 12 (2022) 9308, https://doi.org/10.3390/app12189308.
[111] Z. Deng, Z. Cai, M. Liang, A multi-hop VANETs-assisted offloading strategy in vehicular mobile edge computing, IEEE Access 8 (2020) 53062–53071, https://doi.org/10.1109/ACCESS.2020.2981501.
[112] L. Silva, N. Magaia, B. Sousa, A. Kobusinska, A. Casimiro, C.X. Mavromoustakis, G. Mastorakis, V.H.C. De Albuquerque, Computing paradigms in emerging vehicular environments: a review, IEEE/CAA J. Autom. Sin. 8 (2021) 491–511, https://doi.org/10.1109/JAS.2021.1003862.
[113] L.N.T. Huynh, Q.V. Pham, T.D.T. Nguyen, M.D. Hossain, Y.R. Shin, E.N. Huh, Joint Computational Offloading and Data-Content Caching in NOMA-MEC Networks, IEEE Access 9 (2021) 12943–12954, https://doi.org/10.1109/ACCESS.2021.3051278.
[114] W. Shi, J. Cao, Q. Zhang, Y. Li, L. Xu, Edge computing: vision and challenges, IEEE Internet Things J. 3 (2016) 637–646, https://doi.org/10.1109/JIOT.2016.2579198.
[115] J. Qadir, B. Sainz-De-Abajo, A. Khan, B. García-Zapirain, I. De La Torre-Díez, H. Mahmood, Towards mobile edge computing: taxonomy, challenges, applications and future realms, IEEE Access 8 (2020) 189129–189162, https://doi.org/10.1109/ACCESS.2020.3026938.
[116] X. Xu, Y. Xue, L. Qi, Y. Yuan, X. Zhang, T. Umer, S. Wan, An edge computing-enabled computation offloading method with privacy preservation for internet of connected vehicles, Futur. Gener. Comput. Syst. 96 (2019) 89–100, https://doi.org/10.1016/j.future.2019.01.012.
[117] R.A. Dziyauddin, D. Niyato, N.C. Luong, M.A.M. Izhar, M. Hadhari, S. Daud, Computation offloading and content caching delivery in vehicular edge computing: a survey, ArXiv E-Prints arXiv-1912 (2019). http://arxiv.org/abs/1912.07803.
[118] J. Xu, L. Chen, P. Zhou, Joint service caching and task offloading for mobile edge computing in dense networks, IEEE INFOCOM 2018 - IEEE Conf. Comput. Commun. (2018) 207–215.
[119] D. Han, W. Chen, B. Bai, Y. Fang, Offloading optimization and bottleneck analysis for mobile cloud computing, IEEE Trans. Commun. 67 (2019) 6153–6167, https://doi.org/10.1109/TCOMM.2019.2920348.
[120] W. Junior, E. Oliveira, A. Santos, K. Dias, A context-sensitive offloading system using machine-learning classification algorithms for mobile cloud environment, Futur. Gener. Comput. Syst. 90 (2019) 503–520, https://doi.org/10.1016/j.future.2018.08.026.
[121] C. Jiang, G. Han, J. Lin, G. Jia, W. Shi, J. Wan, Characteristics of co-allocated online services and batch jobs in internet data centers: a case study from alibaba cloud, IEEE Access 7 (2019) 22495–22508, https://doi.org/10.1109/ACCESS.2019.2897898.
[122] Y. Qiu, C. Jiang, Y. Wang, D. Ou, Y. Li, J. Wan, Energy aware virtual machine scheduling in data centers, Energies 12 (2019) 646, https://doi.org/10.3390/en12040646.
[123] C.M. Huang, M.S. Chiang, D.T. Dao, W.L. Su, S. Xu, H. Zhou, V2V data offloading for cellular network based on the software defined network (SDN) inside mobile edge computing (MEC) architecture, IEEE Access 6 (2018) 17741–17755, https://doi.org/10.1109/ACCESS.2018.2820679.
[124] C. Chen, L. Chen, L. Liu, S. He, X. Yuan, D. Lan, Z. Chen, Delay-optimized V2V-based computation offloading in urban vehicular edge computing and networks, IEEE Access 8 (2020) 18863–18873, https://doi.org/10.1109/ACCESS.2020.2968465.
[125] L. Liu, M. Zhao, M. Yu, M.A. Jan, D. Lan, A. Taherkordi, Mobility-aware multi-hop task offloading for autonomous driving in vehicular edge computing and networks, IEEE Trans. Intell. Transp. Syst. (2022), https://doi.org/10.1109/TITS.2022.3142566.
[126] W.U. Khan, F. Jameel, G.A.S. Sidhu, M. Ahmed, X. Li, R. Jantti, Multiobjective optimization of uplink NOMA-enabled vehicle-to-infrastructure communication, IEEE Access 8 (2020) 84467–84478, https://doi.org/10.1109/ACCESS.2020.2991197.
[127] Y. Zhang, J. Lopez, Z. Wang, Mobile edge computing for vehicular networks [from the guest editors], IEEE Veh. Technol. Mag. 14 (2019) 27–108, https://doi.org/10.1109/MVT.2018.2885367.
[128] M.S. Mekala, G. Dhiman, R. Patan, S. Kallam, Deep learning-influenced joint vehicle-to- infrastructure and vehicle-to-vehicle communication approach for internet of vehicles, Expert Syst. 39 (2022) e12815.
[129] Y. Sun, J. Song, S. Zhou, X. Guo, Z. Niu, Task replication for vehicular edge computing: a combinatorial multi-armed bandit based approach, in: 2018 IEEE Glob. Commun. Conf., IEEE, 2018: pp. 1–7. 10.1109/GLOCOM.2018.8647564.
[130] F. Sun, N. Cheng, S. Zhang, H. Zhou, L. Gui, X. Shen, Reinforcement learning based computation migration for vehicular cloud computing, in: 2018 IEEE Glob. Commun. Conf. GLOBECOM, IEEE, 2018: pp. 1–6. 10.1109/GLOCOM.2018.8647996.
[131] Z. Wang, Z. Zhong, M. Ni, Application-aware offloading policy using SMDP in vehicular fog computing systems, in: 2018 IEEE Int. Conf. Commun. Work. ICC Work. 2018 - Proc., IEEE, 2018: pp. 1–6. 10.1109/ICCW.2018.8403696.
[132] Z. Wang, Z. Zhong, D. Zhao, M. Ni, Vehicle-based cloudlet relaying for mobile computation offloading, IEEE Trans. Veh. Technol. 67 (2018) 11181–11191, https://doi.org/10.1109/TVT.2018.2870392.
[133] S.S. Shah, M. Ali, A.W. Malik, M.A. Khan, S.D. Ravana, VFog: a vehicle-assisted computing framework for delay-sensitive applications in smart cities, IEEE Access 7 (2019) 34900–34909, https://doi.org/10.1109/ACCESS.2019.2903302.
[134] Z. Ning, P. Dong, X. Wang, L. Guo, J.J.P.C. Rodrigues, X. Kong, J. Huang, R.Y. K. Kwok, Deep reinforcement learning for intelligent internet of vehicles: an energy-efficient computational offloading scheme, IEEE Trans. Cogn. Commun. Netw. 5 (2019) 1060–1072, https://doi.org/10.1109/TCCN.2019.2930521.
[135] D. Tang, X. Zhang, M. Li, X. Tao, Adaptive inference reinforcement learning for task offloading in vehicular edge computing systems, in: 2020 IEEE Int. Conf. Commun. Work. ICC Work. 2020, IEEE, 2020: pp. 1–6. 10.1109/ICCWorkshops49005.2020.9145133.
[136] J. Feng, Z. Liu, C. Wu, Y. Ji, AVE: autonomous vehicular edge computing framework with ACO-based scheduling, IEEE Trans. Veh. Technol. 66 (2017) 10660–10675, https://doi.org/10.1109/TVT.2017.2714704.
[137] Z. Su, Y. Hui, T.H. Luan, Distributed task allocation to enable collaborative autonomous driving with network softwarization, IEEE J. Sel. Areas Commun. 36 (2018) 2175–2189, https://doi.org/10.1109/JSAC.2018.2869948.
[138] F. Sun, F. Hou, N. Cheng, M. Wang, H. Zhou, L. Gui, X. Shen, Cooperative task scheduling for computation offloading in vehicular cloud, IEEE Trans. Veh. Technol. 67 (2018) 11049–11061, https://doi.org/10.1109/TVT.2018.2868013.
[139] Y. Sun, X. Guo, S. Zhou, Z. Jiang, X. Liu, Z. Niu, Learning-based task offloading for vehicular cloud computing systems, IEEE Int. Conf. Commun., IEEE (2018) 1–7, https://doi.org/10.1109/ICC.2018.8422661.
[140] M. Liwang, S. Dai, Z. Gao, Y. Tang, H. Dai, A truthful reverse-auction mechanism for computation offloading in cloud-enabled vehicular network, IEEE Internet Things J. 6 (2019) 4214–4227, https://doi.org/10.1109/JIOT.2018.2875507.
[141] M. Liwang, S. Dai, Z. Gao, X. Du, M. Guizani, H. Dai, A computation offloading incentive mechanism with delay and cost constraints under 5G satellite-ground IoV architecture, IEEE Wirel. Commun. 26 (2019) 124–132, https://doi.org/10.1109/MWC.2019.1800364.
[142] Y. Sun, X. Guo, J. Song, S. Zhou, Z. Jiang, X. Liu, Z. Niu, Adaptive learning-based task offloading for vehicular edge computing systems, IEEE Trans. Veh. Technol. 68 (2019) 3061–3074, https://doi.org/10.1109/TVT.2019.2895593.
[143] J. Shi, J. Du, J. Wang, J. Yuan, Distributed V2V computation offloading based on dynamic pricing using deep reinforcement learning, 2020 IEEE Wirel. Commun. Netw. Conf. (2020) 1–6. 10.1109/WCNC45663.2020.9120816.
[144] H. Liang, X. Zhang, X. Hong, Z. Zhang, M. Li, G. Hu, F. Hou, Reinforcement learning enabled dynamic resource allocation in the internet of vehicles, IEEE Trans. Ind. Informatics 17 (2021) 4957–4967, https://doi.org/10.1109/TII.2020.3019386.
[145] K. Xiong, S. Leng, X. Chen, C. Huang, C. Yuen, Y.L. Guan, Communication and computing resource optimization for connected autonomous driving, IEEE Trans. Veh. Technol. 69 (2020) 12652–12663, https://doi.org/10.1109/TVT.2020.3029109.
[146] J. Zhou, D. Tian, Y. Wang, Z. Sheng, X. Duan, V.C.M. Leung, Reliability-oriented optimization of computation offloading for cooperative vehicle-infrastructure systems, IEEE Signal Process. Lett. 26 (2019) 104–108, https://doi.org/10.1109/LSP.2018.2880081.
[147] J. Zhao, Q. Li, Y. Gong, K. Zhang, Computation offloading and resource allocation for cloud assisted mobile edge computing in vehicular networks, IEEE Trans. Veh. Technol. 68 (2019) 7944–7956, https://doi.org/10.1109/tvt.2019.2917890.
[148] H. Peng, X. Shen, Multi-agent reinforcement learning based resource management in MEC- and UAV-assisted vehicular networks, IEEE J. Sel. Areas Commun. 39 (2021) 131–141, https://doi.org/10.1109/JSAC.2020.3036962.
[149] Q. Ye, W. Shi, K. Qu, H. He, W. Zhuang, X. Shen, Joint RAN slicing and computation offloading for autonomous vehicular networks: a learning-assisted hierarchical approach, IEEE Open J. Veh. Technol. 2 (2021) 272–288, https://doi.org/10.1109/OJVT.2021.3089083.
[150] H. Wang, X. Li, H. Ji, H. Zhang, Dynamic offloading scheduling scheme for MEC-enabled vehicular networks, in: 2018 IEEE/CIC Int. Conf. Commun. China (ICCC Work., IEEE, 2018: pp. 206–210. 10.1109/ICCChinaW.2018.8674508.
[151] Z. Zhou, P. Liu, Z. Chang, C. Xu, Y. Zhang, Energy-efficient workload offloading and power control in vehicular edge computing, in: 2018 IEEE Wirel. Commun. Netw. Conf. Work., IEEE, 2018: pp. 191–196. 10.1109/WCNCW.2018.8368975.
[152] X. Huang, K. Xu, C. Lai, Q. Chen, J. Zhang, Energy-efficient offloading decision-making for mobile edge computing in vehicular networks, EURASIP J. Wirel. Commun. Netw. 1 (2020) 1–16, https://doi.org/10.1186/s13638-020-1652-5.
[153] H. Ke, J. Wang, L. Deng, Y. Ge, H. Wang, Deep reinforcement learning-based adaptive computation offloading for MEC in heterogeneous vehicular networks, IEEE Trans. Veh. Technol. 69 (2020) 7916–7929, https://doi.org/10.1109/TVT.2020.2993849.
[154] W. Zhan, C. Luo, J. Wang, C. Wang, G. Min, H. Duan, Q. Zhu, Deep-reinforcement-learning-based offloading scheduling for vehicular edge computing, IEEE Internet Things J. 7 (2020) 5449–5465, https://doi.org/10.1109/JIOT.2020.2978830.
[155] S. Raza, S. Wang, M. Ahmed, M.R. Anwar, M.A. Mirza, W.U. Khan, Task offloading and resource allocation for IoV using 5G NR-V2X communication, IEEE Internet Things J. 9 (2021) 10397–10410, https://doi.org/10.1109/JIOT.2021.3121796.
[156] Y. Ouyang, Task offloading algorithm of vehicle edge computing environment based on Dueling-DQN, J. Phys. Conf. Ser., IOP Publishing (2021) 012046, https://doi.org/10.1088/1742-6596/1873/1/012046.
[157] K. Zhang, Y. Mao, S. Leng, A. Vinel, Y. Zhang, Delay constrained offloading for mobile edge computing in cloud-enabled vehicular networks, in: IEEE (Ed.), 2016 8th Int. Work. Resilient Networks Des. Model., 2016: pp. 288–294. 10.1109/RNDM.2016.7608300.
[158] W. Tang, S. Li, W. Rafique, W. Dou, S. Yu, An offloading approach in fog computing environment, in: 2018 IEEE SmartWorld, Ubiquitous Intell. Comput. Adv. Trust. Comput. Scalable Comput. Commun. Cloud Big Data Comput. Internet

People Smart City Innov. (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)/ScalCom/CBDC, IEEE, 2018: pp. 857–864. 10.1109/SmartWorld.2018.00157.
[159] S. Wu, W. Xia, W. Cui, Q. Chao, Z. Lan, F. Yan, L. Shen, An efficient offloading algorithm based on support vector machine for mobile edge computing in vehicular networks, in: 2018 10th Int. Conf. Wirel. Commun. Signal Process., IEEE, 2018: pp. 1–6. 10.1109/WCSP.2018.8555695.
[160] J. Zhou, F. Wu, K. Zhang, Y. Mao, S. Leng, Joint optimization of offloading and resource allocation in vehicular networks with mobile edge computing, in: 2018 10th Int. Conf. Wirel. Commun. Signal Process., IEEE, 2018: pp. 1–6. 10.3772/j.issn.1002-0470.2020.08.001.
[161] Y. Liu, S. Wang, J. Huang, F. Yang, A computation offloading algorithm based on game theory for vehicular edge networks, in: 2018 IEEE Int. Conf. Commun., IEEE, 2018: pp. 1–6. 10.1109/ICC.2018.8422240.
[162] K. Zhang, Y. Mao, S. Leng, S. Maharjan, A. Vinel, Y. Zhang, Contract-theoretic approach for delay constrained offloading in vehicular edge computing networks, Mob. Networks Appl. 24 (2019) 1003–1014, https://doi.org/10.1007/s11036-018-1032-0.
[163] X. Lin, J. Li, W. Yang, J. Wu, Z. Zong, X. Wang, Vehicle-to-cloudlet: game-based computation demand response for mobile edge computing through vehicles, in: 2019 IEEE 89th Veh. Technol. Conf., IEEE, 2019: pp. 1–6. 10.1109/VTCSpring.2019.8746335.
[164] C. Yang, Y. Liu, X. Chen, W. Zhong, S. Xie, Efficient mobility-aware task offloading for vehicular edge computing networks, IEEE Access 7 (2019) 26652–26664, https://doi.org/10.1109/ACCESS.2019.2900530.
[165] Y. Cui, Y. Liang, R. Wang, Resource allocation algorithm with multi-platform intelligent offloading in D2D-enabled vehicular networks, IEEE Access 7 (2019) 21246–21253, https://doi.org/10.1109/ACCESS.2018.2882000.
[166] P. Liu, J. Li, Z. Sun, Matching-based task offloading for vehicular edge computing, IEEE Access 7 (2019) 27628–27640, https://doi.org/10.1109/ACCESS.2019.2896000.
[167] V.D. Nguyen, T.T. Khanh, N.H. Tran, E.N. Huh, C.S. Hong, Joint offloading and IEEE 802.11p-based contention control in vehicular edge computing, IEEE Wirel. Commun. Lett. 9 (2020) 1014–1018, https://doi.org/10.1109/LWC.2020.2978482.
[168] M. Li, J. Gao, L. Zhao, X. Shen, Deep reinforcement learning for collaborative edge computing in vehicular networks, IEEE Trans. Cogn. Commun. Netw. 6 (2020) 1122–1135, https://doi.org/10.1109/TCCN.2020.3003036.
[169] J. Sun, Q. Gu, T. Zheng, P. Dong, A. Valera, Y. Qin, Joint optimization of computation offloading and task scheduling in vehicular edge computing networks, IEEE Access 8 (2020) 10466–10477, https://doi.org/10.1109/ACCESS.2020.2965620.
[170] M. Khayyat, I.A. Elgendy, A. Muthanna, A.S. Alshahrani, S. Alharbi, A. Koucheryavy, Advanced deep learning-based computational offloading for multilevel vehicular edge-cloud computing networks, IEEE Access 8 (2020) 137052–137062, https://doi.org/10.1109/ACCESS.2020.3011705.
[171] Z. Ning, Y. Li, P. Dong, X. Wang, M.S. Obaidat, X. Hu, L. Guo, Y. Guo, J. Huang, B. Hu, When deep reinforcement learning meets 5G-enabled vehicular networks: a distributed offloading framework for traffic big data, IEEE Trans. Ind. Informatics 16 (2020) 1352–1361, https://doi.org/10.1109/TII.2019.2937079.
[172] K. Zhang, S. Maharjan, Y. Zhang, Optimal delay constrained offloading for vehicular edge computing networks, in: 2017 IEEE Int. Conf. Commun., 2017: pp. 1–6. 10.1109/MoWNet.2018.8428935.
[173] M. Pasha, K.U.R. Khan, Opportunistic task offloading in vehicular networks, in: Proc. 3rd IEEE Int. Conf. Adv. Electr. Electron. Information, Commun. Bio-Informatics, AEEICB 2017, IEEE, 2017: pp. 510–514. 10.1109/AEEICB.2017.7972366.
[174] Z. Jiang, S. Zhou, X. Guo, Z. Niu, Task replication for deadline-constrained vehicular cloud computing: optimal policy performance analysis, and implications on road traffic, IEEE Internet Things J. 5 (2018) 93–107, https://doi.org/10.1109/JIOT.2017.2771473.
[175] G. Qiao, S. Leng, K. Zhang, Y. He, Collaborative task offloading in vehicular edge multi-access networks, IEEE Commun. Mag. 56 (2018) 48–54, https://doi.org/10.1109/MCOM.2018.1701130.
[176] H. Wang, X. Li, H. Ji, H. Zhang, Federated offloading scheme to minimize latency in MEC-enabled vehicular networks, in: 2018 IEEE Globecom Work. (GC Wkshps), IEEE, 2019: pp. 1–6. 10.1109/GLOCOMW.2018.8644315.
[177] X. Xu, Y. Xue, X. Li, L. Qi, S. Wan, A computation offloading method for edge computing with vehicle-to-everything, IEEE Access 7 (2019) 131068–131077, https://doi.org/10.1109/ACCESS.2019.2940295.
[178] Q. Qi, J. Wang, Z. Ma, H. Sun, Y. Cao, L. Zhang, J. Liao, Knowledge-driven service offloading decision for vehicular edge computing: a deep reinforcement learning approach, IEEE Trans. Veh. Technol. 68 (2019) 4192–4203, https://doi.org/10.1109/TVT.2019.2894437.
[179] L. Wang, Q. Zhang, Y. Li, H. Zhong, W. Shi, MobileEdge: Enhancing on-board vehicle computing units using mobile edges for CAVs, in: 2019 IEEE 25th Int. Conf. Parallel Distrib. Syst., IEEE, 2019: pp. 470–479. 10.1109/ICPADS47876.2019.00073.
[180] X. Hou, Z. Ren, J. Wang, W. Cheng, Y. Ren, K.C. Chen, H. Zhang, Reliable computation offloading for edge-computing-enabled software-defined IoV, IEEE Internet Things J. 7 (2020) 7097–7111, https://doi.org/10.1109/JIOT.2020.2982292.
[181] S. Raza, W. Liu, M. Ahmed, M.R. Anwar, M.A. Mirza, Q. Sun, S. Wang, An efficient task offloading scheme in vehicular edge computing, J. Cloud Comput. 9 (2020) 1–14, https://doi.org/10.1186/s13677-020-00175-w.
[182] H. Maleki, M. Basaran, L. Durak-Ata, Reinforcement learning-based decision-making for vehicular edge computing, in: 2021 29th Signal Process. Commun. Appl. Conf., IEEE, 2021: pp. 1–4. 10.1109/SIU53274.2021.9478026.
[183] X. Hou, Y. Li, M. Chen, D. Wu, D. Jin, S. Chen, Vehicular fog computing: a viewpoint of vehicles as the infrastructures, IEEE Trans. Veh. Technol. 65 (2016) 3860–3873, https://doi.org/10.1109/TVT.2016.2532863.
[184] K. Zhang, Y. Mao, S. Leng, Y. He, Y. Zhang, Predictive offloading in cloud-driven vehicles: using mobile-edge computing for a promising network paradigm, IEEE Veh. Technol. Mag. 12 (2017) 36–44.
[185] Q. Liu, Z. Su, Y. Hui, Computation offloading scheme to improve QoE in vehicular networks with mobile edge computing, in: 2018 10th Int. Conf. Wirel. Commun. Signal Process. WCSP 2018, IEEE, 2018: pp. 1–5. 10.1109/WCSP.2018.8555879.
[186] L.T. Tan, R.Q. Hu, Mobility-aware edge caching and computing in vehicle networks: a deep reinforcement learning, IEEE Trans. Veh. Technol. 67 (2018) 10190–10203, https://doi.org/10.1109/TVT.2018.2867191.
[187] X. Wang, Z. Ning, L. Wang, Offloading in internet of vehicles: a fog-enabled real-time traffic management system, IEEE Trans. Ind. Informatics 14 (2018) 4568–4578, https://doi.org/10.1109/TII.2018.2816590.
[188] L. Li, H. Zhou, S.X. Xiong, J. Yang, Y. Mao, Compound model of task arrivals and load-aware offloading for vehicular mobile edge computing networks, IEEE Access 7 (2019) 26631–26640, https://doi.org/10.1109/ACCESS.2019.2901280.
[189] S. Mu, Z. Zhong, M. Ni, Multi-Destination computation offloading in vehicular networks, in: 2018 14th Int. Wirel. Commun. Mob. Comput. Conf., IEEE, 2018: pp. 446–451. 10.1109/IWCMC.2018.8450308.
[190] J. Zhao, L. Wang, K.-K. Wong, M. Tao, T. Mahmoodi, Energy and latency control for edge computing in dense V2X networks, ArXiv Prepr. ArXiv1807.02311 (2018). http://arxiv.org/abs/1807.02311.
[191] X. He, H. Lu, M. Du, Y. Mao, K. Wang, QoE-based task offloading with deep reinforcement learning in edge-enabled internet of vehicles, IEEE Trans. Intell. Transp. Syst. 22 (2021) 2252–2261, https://doi.org/10.1109/TITS.2020.3016002.
[192] Q. Yuan, J. Li, Z. Liu, F. Yang, Space and time constrained data offloading in vehicular networks, in: 2016 IEEE 18th Int. Conf. High Perform. Comput. Commun. IEEE 14th Int. Conf. Smart City; IEEE 2nd Int. Conf. Data Sci. Syst., IEEE, 2016. 10.1109/HPCC-SmartCity-DSS.2016.188.
[193] J. Liu, J. Wan, B. Zeng, Q. Wang, H. Song, M. Qiu, A scalable and quick-response software defined vehicular network assisted by mobile edge computing, IEEE Commun. Mag. 55 (2017) 94–100, https://doi.org/10.1109/MCOM.2017.1601150.
[194] Q. Qi, Z. Ma, Vehicular edge computing via deep reinforcement learning, ArXiv Prepr. ArXiv (2018) 1901.04290. http://arxiv.org/abs/1901.04290.
[195] S. Wan, X. Li, Y. Xue, W. Lin, X. Xu, Efficient computation offloading for Internet of Vehicles in edge computing-assisted 5G networks, J. Supercomput. 76 (2020) 2518–2547, https://doi.org/10.1007/s11227-019-03011-4.
[196] L. Li, T. Lv, P. Huang, P.T. Mathiopoulos, Cost optimization of partial computation offloading and pricing in vehicular networks, J. Signal Process. Syst. 92 (2020) 1421–1435, https://doi.org/10.1007/s11265-020-01572-9.
[197] Z. Ullah, F. Al-Turjman, L. Mostarda, R. Gagliardi, Applications of artificial intelligence and machine learning in smart cities, Comput. Commun. 154 (2020) 313–323, https://doi.org/10.1007/978-3-030-29022-1_1.
[198] B.L. Nguyen, D.T. Ngo, M.N. Dao, V.N.Q. Bao, H.L. Vu, Scheduling and power control for connectivity enhancement in multi-hop I2V/V2V networks, IEEE Trans. Intell. Transp. Syst. (2021), https://doi.org/10.1109/TITS.2021.3091130.
[199] A. Adiththan, S. Ramesh, S. Samii, Cloud-assisted control of ground vehicles using adaptive computation offloading techniques, in: 2018 Des. Autom. Test Eur. Conf. Exhib. DATE, IEEE, 2018: pp. 589–592. 10.23919/DATE.2018.8342076.
[200] L. Yao, X. Xu, M. Bilal, H. Wang, Dynamic edge computation offloading for internet of vehicles with deep reinforcement learning, IEEE Trans. Intell. Transp. Syst. 24 (2023) 12991–12999, https://doi.org/10.1109/TITS.2022.3178759.
[201] X. Xu, X. Zhang, X. Liu, J. Jiang, L. Qi, M.Z.A. Bhuiyan, Adaptive computation offloading with edge for 5G-envisioned internet of connected vehicles, IEEE Trans. Intell. Transp. Syst. 22 (2021) 5213–5222, https://doi.org/10.1109/TITS.2020.2982186.
[202] P. Sun, N. Aljeri, A. Boukerche, Machine learning-based models for real-time traffic flow prediction in vehicular networks, IEEE Netw. 34 (2020) 178–185, https://doi.org/10.1109/MNET.011.1900338.
[203] K. Tan, D. Bremner, J. Le Kernec, L. Zhang, M. Imran, Machine learning in vehicular networking: an overview, Digit. Commun. Networks 8 (2022) 18–24, https://doi.org/10.1016/j.dcan.2021.10.007.
[204] T. Liu, L. Tang, W. Wang, X. He, Q. Chen, X. Zeng, H. Jiang, Resource allocation in DT-assisted internet of vehicles via edge intelligent cooperation, IEEE Internet Things J. 9 (2022) 17608–17626, https://doi.org/10.1109/JIOT.2022.3156100.
[205] B. Shen, X. Xu, L. Qi, X. Zhang, G. Srivastava, Dynamic server placement in edge computing toward Internet of Vehicles, Comput. Commun. 178 (2021) 114–123, https://doi.org/10.1016/j.comcom.2021.07.021.
[206] M. Aazam, S. Zeadally, E.F. Flushing, Task offloading in edge computing for machine learning-based smart healthcare, Comput. Networks 191 (2021) 108019, https://doi.org/10.1016/j.comnet.2021.108019.
[207] G. Qu, H. Wu, R. Li, P. Jiao, DMRO: a deep meta reinforcement learning-based task offloading framework for edge-cloud computing, IEEE Trans. Netw. Serv. Manag. 18 (2021) 3448–3459, https://doi.org/10.1109/TNSM.2021.3087258.
[208] P. Agbaje, A. Anjum, A. Mitra, E. Oseghale, G. Bloom, H. Olufowobi, Survey of interoperability challenges in the internet of vehicles, IEEE Trans. Intell. Transp. Syst. 23 (2022) 22838–22861, https://doi.org/10.1109/TITS.2022.3194413.
[209] J. Zhao, X. Sun, Q. Li, X. Ma, Edge caching and computation management for real-time internet of vehicles: an online and distributed approach, IEEE Trans. Intell.

Transp. Syst. 22 (2020) 2183–2197, https://doi.org/10.1109/TITS.2020.3012966.
[210] X. Zheng, M. Li, Y. Chen, J. Guo, M. Alam, W. Hu, Blockchain-based secure computation offloading in vehicular networks, IEEE Trans. Intell. Transp. Syst. 22 (2021) 4073–4087, https://doi.org/10.1109/TITS.2020.3014229.
[211] Y. Liu, H. Yu, S. Xie, Y. Zhang, Deep reinforcement learning for offloading and resource allocation in vehicle edge computing and networks, IEEE Trans. Veh. Technol. 68 (2019) 11158–11168, https://doi.org/10.1109/TVT.2019.2935450.
[212] Y. Dai, D. Xu, S. Maharjan, G. Qiao, Y. Zhang, Artificial intelligence empowered edge computing and caching for internet of vehicles, IEEE Wirel. Commun. 26 (2019) 12–18, https://doi.org/10.1109/MWC.2019.1800411.
[213] Z. Zhang, Y. Yang, M. Hua, C. Li, Y. Huang, L. Yang, Proactive caching for vehicular multi-view 3D video streaming via deep reinforcement learning, IEEE Trans. Wirel. Commun. 18 (2019) 2693–2706, https://doi.org/10.1109/TWC.2019.2907077.
[214] G. Qiao, S. Leng, S. Maharjan, Y. Zhang, N. Ansari, Deep reinforcement learning for cooperative content caching in vehicular edge computing and networks, IEEE Internet Things J. 7 (2019) 247–257, https://doi.org/10.1109/JIOT.2019.2945640.
[215] L. Liang, H. Ye, G.Y. Li, Spectrum sharing in vehicular networks based on multi-agent reinforcement learning, IEEE J. Sel. Areas Commun. 37 (2019) 2282–2292, https://doi.org/10.1109/JSAC.2019.2933962.
[216] T. Lu, S. Chang, W. Li, Fog computing enabling geographic routing for urban area vehicular network, Peer-to-Peer Netw. Appl. 11 (2018) 749–755, https://doi.org/10.1007/s12083-017-0560-x.
[217] Z. Zhang, W. Zhao, O. Huang, G. Jia, Y. Li, S. Mei, D. Zhao, Edge network routing protocol base on target tracking scenario, Mob. Networks Appl. 26 (2021) 2230–2241.
[218] S. Misra, N. Saha, Detour: dynamic task offloading in software-defined fog for IoT applications, IEEE J. Sel. Areas Commun. 37 (2019) 1159–1166, https://doi.org/10.1109/JSAC.2019.2906793.
[219] J. Huang, Y. Qian, R.Q. Hu, Secure and efficient privacy-preserving authentication scheme for 5G software defined vehicular networks, IEEE Trans. Veh. Technol. 69 (2020) 8542–8554, https://doi.org/10.1109/TVT.2020.2996574.
[220] M. Tao, K. Ota, M. Dong, Foud: integrating fog and cloud for 5G-enabled V2G networks, IEEE Netw. 31 (2017) 8–13, https://doi.org/10.1109/MNET.2017.1600213NM.
[221] X. Chen, L. Wang, Exploring fog computing-based adaptive vehicular data scheduling policies through a compositional formal method - PEPA, IEEE Commun. Lett. 21 (2017) 745–748, https://doi.org/10.1109/LCOMM.2016.2647595.
[222] N. Iqbal, A.N. Khan, A. Imran, F. Rizwan, S. Qayyum, R. Malik, D.H.K. Ahmad, Enhanced time-constraint aware tasks scheduling mechanism based on predictive optimization for efficient load balancing in smart manufacturing, J. Manuf. Syst. 64 (2022) 19–39, https://doi.org/10.1016/j.jmsy.2022.05.015.
[223] S.A. Darade, M. Akkalakshmi, N. Pagar, SDN based load balancing technique in internet of vehicle using integrated whale optimization method SDN based load balancing technique in internet of vehicle using integrated whale optimization method, AIP Conf. Proc. 2469 (2022) 020006.
[224] S. Lee, S. Lee, Resource allocation for vehicular fog computing using reinforcement learning combined with heuristic information, IEEE Internet Things J. 7 (2020) 10450–10464, https://doi.org/10.1109/JIOT.2020.2996213.
[225] M. Ouaissa, M. Ouaissa, M. Houmer, S. El Hamdani, Z. Boulouard, A secure vehicle to everything (V2X) communication model for intelligent transportation system, in: Comput. Intell. Recent Commun. Networks, Springer, Cham., 2022: pp. 83–102. 10.1007/978-3-030-77185-0_6.
[226] H. Alsulami, S.H. Serbaya, E.H. Abualsauod, A.M. Othman, A. Rizwan, A. Jalali, A Federated deep learning empowered resource management method to optimize 5G and 6G quality of services (QoS), Wirel. Commun. Mob. Comput. 2022 (2022), https://doi.org/10.1155/2022/1352985.
[227] E. Qafzezi, K. Bylykbashi, P. Ampririt, M. Ikeda, K. Matsuo, L. Barolli, An intelligent approach for cloud-fog-edge computing SDN-VANETs based on fuzzy logic: effect of different parameters on coordination and management of resources, Sensors 22 (2022) 878, https://doi.org/10.3390/s22030878.
[228] I. Acm, J. Li, C. Natalino, D.P. Van, L. Wosinska, J. Chen, Resource management in fog-enhanced radio access network to support real-time vehicular services, in: IEEE 1st Int. Conf. Fog Edge Comput., IEEE, 2017: pp. 68–74. 10.1109/ICFEC.2017.17.
[229] S. Basudan, X. Lin, K. Sankaranarayanan, A privacy-preserving vehicular crowdsensing-based road surface condition monitoring system using fog computing, IEEE Internet Things J. 4 (2017) 772–782, https://doi.org/10.1109/JIOT.2017.2666783.
[230] K. Fan, J. Wang, X. Wang, H. Li, Y. Yang, Secure, efficient and revocable data sharing scheme for vehicular fogs, Peer-to-Peer Netw. Appl. 11 (2018) 766–777, https://doi.org/10.1007/s12083-017-0562-8.
[231] S.A. Soleymani, A.H. Abdullah, A secure trust model based on fuzzy logic in vehicular Ad Hoc networks with fog computing, IEEE Access 5 (2017) 15619–15629.
[232] J.P.D. Comput, S. Wang, Y. Zhao, L. Huang, J. Xu, C. Hsu, QoS prediction for service recommendations in mobile edge computing, J. Parallel Distrib. Comput. 127 (2019) 134–144, https://doi.org/10.1016/j.jpdc.2017.09.014.
[233] A. Belhadi, Y. Djenouri, G. Srivastava, D. Djenouri, J.-C.-W. Lin, G. Fortino, Deep learning for pedestrian collective behavior analysis in smart cities: a model of group trajectory outlier detection, Inf. Fusion 65 (2021) 13–20, https://doi.org/10.1016/j.econlet.2020.109008.
[234] J. Liu, M. Ahmed, M. Ayzed Mirza, W. Ullah Khan, D. Xu, J. Li, A. Aziz, Z. Han, RL / DRL meets vehicular task offloading using edge and vehicular cloudlet : a survey, IEEE Internet Things J. 9 (2022) 8315–8338.

Marieh Talebkhah, has received the B.Sc. of Computer Software Engineering from Iran, in 2009, and M.Sc. degree in Computer Software Engineering from Iran, in 2017. She is currently a Ph.D. candidate in Universiti Putra Malaysia (UPM), Kuala Lumpur. Her current research focuses on vehicular communication, intelligent transportation system, wireless ad hoc network, IoT, cloud and Fog computing, and big data analytics.

Aduwati Sali, received the B.Eng. degree in electrical electronics engineering from The University of Edinburgh, U.K., in 1999, the M.Sc. degree in communications and network engineering from Universiti Putra Malaysia, in 2002, and the Ph.D. degree in mobile and satellite communications from the University of Surrey, U.K., in 2009. She was the Deputy Director of the Research Management Centre (RMC), Universiti Putra Malaysia, from 2016 to 2019. She is currently a Professor with the Department of Computer and Communication System Engineering and a Researcher with the Wireless and Photonic Network Research Center of Excellence (WiPNET), Universiti Putra Malaysia. Her research interests include mobile and satellite communication systems.

Vahid Khodamoradi, received the B.S. degree in computer engineering from Payame Noor University in Ilam, Tehran, Iran, in 2009. He also earned an M.S. degree in computer science from Universiti Putra Malaysia (UPM), Seri Kembangan, Malaysia, in 2014, and a Ph.D. degree in wireless communications and network engineering from the Department of Computer and Communication Systems Engineering, Faculty of Engineering at UPM. His research interests include green wireless communications, energy efficiency optimization, convex and non-convex optimization, massive MIMO systems and hybrid networks, and wireless powered communications.

Touraj Khodadadi is a faculty member of the School of Science and Engineering at Malaysia University of Science and Technology (MUST). He earned his Ph.D. and M.Sc. in Information Security from Universiti Teknologi Malaysia (UTM) in Malaysia. His teaching interests encompass a broad range of subjects, including Cyber and Network Security, Authentication Systems, Cloud Computing, Big Data, and Machine Learning. In addition, he has authored and co-authored over 40 international journal articles and conference papers, delving into various aspects of computer science, information, and network security. His primary research interests revolve around authentication systems, both in the context of wireless and wired network security. Furthermore, he specializes in cryptography, graphical passwords, network authentication, data mining, artificial intelligence, and cloud computing. Beyond his responsibilities in teaching and research at the university, he has also contributed as an editor and reviewer for several international journals and conferences. Additionally, he is a member of various review panels responsible for evaluating master's and doctoral research defenses.

Meisam Gordan is currently working on a European Union's Horizon 2020 funded ongoing project, named PRECINCT as a Postdoctoral Research Fellow at University College Dublin (UCD), Ireland. He received the Ph.D. degree in the field of structural engineering and materials with a focus on structural health monitoring, data mining, and artificial intelligence from the University of Malaya, in 2020. He has been recognized by several peer-reviewed journals as a reliable referee with over 20 peer-reviewed articles and being an editorial board member of several peer-reviewed journals. He has published over 20 articles in peer-reviewed scientific international journals in different fields, such as Structural Health Monitoring, Signal Processing, Data Mining, Artificial Intelligence, Soft Computing, Vibration Control Systems, IoT, Big Data, Blockchain, Remote Sensing, Smart Cities, Unmanned Aerial Systems, Finite Element Analysis, and Materials. His research interests include the connection between structural engineering and computer science using industry 4.0 emerging technologies. He is a member of renowned institutes, such as the American Society of Civil Engineers (ASCE), Board of Engineers Malaysia (BEM), and Institution of Engineers Malaysia (IEM).