Li *et al. J Wireless Com Network* (2024) 2024:13
https://doi.org/10.1186/s13638-024-02341-z

EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

An adaptive transmission strategy based on cloud computing in IoV architecture

Bin Li[1,4], Vivian Li[2†], Miao Li[1], John Li[3], Jiaqi Yang[4] and Bin Li[4*]

†Bin Li and Vivian Li contributed equally to this work.

*Correspondence: benliinz@gmail.com

[1] School of Communication and Media, Guangzhou Huashang College, Guangzhou 511300, China
[2] Faculty of Science, The University of Auckland, Auckland 1052, New Zealand
[3] Faculty of Medical and Health Science, The University of Auckland, Auckland 1052, New Zealand
[4] Faculty of Humanities, ZhuHai City Polytechnic, Zhuhai 519090, China

Abstract

Because of recent developments in wireless communication, sensor technology, and computing technology, researchers have recently shown a significant amount of interest in the Internet of Vehicles (IoV), which has become feasible as a result of these improvements. Because of the distinctive characteristics of IoV, such as the varied compute and communication capacities of network nodes, it is difficult to process jobs that are time-sensitive. The purpose of this study is to investigate the ways in which cloud computing may collaborate with the IoV to make the processing of time-sensitive procedures easier. We propose a vehicle design that makes advantage of cloud computing as a means of accomplishing this goal. Increasing the proportion of time-sensitive jobs that are ultimately completed was the motivation behind the development of the offloading model that we devised. Taking this into perspective, we present an adaptive task offloading and transmission method. Taking into account the ever-changing requirements and constraints on the available resources, this algorithm dynamically organizes all of the tasks into separate cloud link lists on the cloud. Following that, the tasks contained within each list are distributed in a cooperative manner to a number of different nodes, with the characteristics of those nodes being taken into consideration. Following the presentation of the simulation model, we carried out an experimental investigation into the effectiveness of the model that was proposed. It is abundantly evident that the proposed model is effective, as indicated by the findings.

Keywords: Cloud computing, Cloud nodes, Computational delay, Internet of Vehicles, Resource utilization, Task offloading, Transmission rate

1 Introduction

Due to the fast growth of wireless communication, the Internet of Vehicles (IoV) has developed as a significant application of the Internet of Things (IoT) [1]. The Internet of Vehicles (IoV) is a dynamic mobile communication system in which vehicles interact with one another and roadside units, which are termed vehicle access points (VAPs), over public networks to improve the connection between them [2]. IoV has developed enormous capacities to make urban traffic safer and smarter as a result of its wide variety of entertainment services, fleet operations, and in-vehicle application options [3]. IoV is now a topic of widespread concern in both the academic world and the business world as a result of the widespread deployment of sensors and the fast development of

cognitive technology. Massive task demands are sent to the VAPs in the IoV, where they are then analyzed in real time for the purpose of enhancing the driving and traveling experience [4]. However, in order to accomplish vehicle-to-everything (V2X) communications with a minimum amount of lag time, the mobility of the ever-increasing number of cars presents a significant issue.

It has become clear that the next generation of wireless networks is going to be an essential component in meeting the rigorous connection requirements of IoV [5]. When compared to 4G, 5G is distinguished by its high bandwidth and low latency, both of which work together to dramatically enhance the quality of experience (QoE) for consumers [6]. The information that is produced by the cars is sent to a remote cloud for storage and processing through the wireless network [7]. This is a contributing factor and helps to ensure that the enormous services that IoV offers continue to function properly. On the other hand, due to the enormous physical distance that exists between the base stations (BSs) and the cloud, it might be difficult to transmit back the results of the task requests in IoV in a timely manner [8]. A high-performance computing paradigm that is based on cloud computing is employed so that automobiles may enjoy high-quality services in real time. This is done in order to deliver real-time services for the applications that are associated with vehicles. Cloud computing provides access to a vast array of computing resources, allowing for the relocation of task executions to computer nodes that are physically placed in close proximity to the end devices. This results in a considerable decrease in the latency that is associated with task offloading as well as transmission delay [9, 10]. In particular, the Internet of Cars (IoV) promotes VAPs and BSs as cloud nodes so that they can accommodate the computational activities and application data coming from the cars that they cover. To do this, we use the VAPs and BSs as edge nodes in the network. Not only does short-distance offloading in cloud computing save time, but it also protects against attacks that could happen with traditional transmission, which ultimately improves data integrity and security [11]. Short-distance offloading also offers the benefit of offloading delay.

On the other hand, the hybrid nodes that are present in both VAPs and BSs make the destinations of the tasks more difficult to determine.

Therefore, figuring out how to identify the destinations of the computing tasks that are being offloaded remains a difficult challenge in IoV. During the process of transmission and offloading, the resource utilization of the edge nodes should also be viewed as a vital indicator since it has an influence on both the overall implementation efficiency of the tasks and the operational performance of the edge nodes. This is because it has an impact on both of these aspects. This is because of the relationship between the two factors. Therefore, in this paper, an adaptive computational offloading and transmission approach for the Internet of Vehicles (IoV) is developed. The purpose of this method is to optimize both transmission latency and resource utilization.

The proposed adaptive transmission approach can be integrated with fifth-generation wireless technology and edge computing. Implementing an adaptive transmission approach for real-time video streaming might be accomplished by a mobile operator with the help of 5G and edge computing. It may be possible for the adaptive transmission technique to make use of 5G in order to send low-latency video streams of high quality to mobile devices. The processing of video streams might be offloaded from the

cloud to the edge of the network via edge computing. This would result in improved performance and reduced backhaul traffic. The implementation of an adaptive transmission technique for traffic management in a smart city might be possible with the help of 5G and edge computing. The adaptive transmission technique may take advantage of 5G to collect and send in real-time traffic data derived from sensors and cameras. Computing at the edge could be utilized to process the data on the traffic and generate recommendations for the regulation of the traffic. The adaptive transmission technique might then make use of 5G in order to convey the recommendations to the various devices, including traffic lights. An adaptive transmission method for industrial automation might be used by a manufacturing organization with the help of 5G and edge computing. Data collection and transmission from industrial sensors and actuators might be done in real time with the help of the adaptive transmission technique and 5G. The data could be processed with edge computing, and control signals could be generated for the industrial machinery using those signals. After that, the adaptive transmission technique might make use of 5G to provide control messages to the industrial equipment. 5G and edge computing present a number of opportunities that can help improve the performance of the suggested adaptive transmission approach as well as the applicability of the strategy. It is conceivable to develop new and innovative applications that have the potential to improve the lives of people and businesses all over the world if an adaptive transmission approach is combined with 5G and edge computing.

The adaptive transmission method can be used to improve the performance of telecommunications networks in a variety of different ways, such as by boosting throughput, reducing latency, and improving dependability. These are just some of the ways in which this can be accomplished. For instance, dynamic bitrate adaptation techniques for video streaming and other applications can be implemented with the help of the adaptive transmission method. Technology like the adaptive transmission method can be utilized to better bring media material to consumers. It achieves this by altering the transmitted signal so that it can be received by the target device. This includes the media and entertainment sectors. Adaptive streaming systems for video and audio data, for instance, may make use of the adaptive transmission approach. The data can be transmitted online via these systems.

Adaptive transmission is becoming increasingly popular because it has potential entertainment applications in the gaming industry. For instance, the adaptive transmission approach can be used in online games to reduce both the amount of delay that occurs and the number of packets that are dropped during transmission. When applied to the realm of education, the adaptive transmission method has the potential to be exploited as a means of facilitating an improvement in the dissemination of instructional content to students. The approach of adaptive transmission can be applied to put adaptive streaming tactics for video lectures and other types of instructional information into action. The adaptive transmission strategy has the ability to dramatically improve both the quality of treatment provided to patients and their level of contentment within the context of the healthcare industry. For example, the adaptive transmission method could be implemented in the building of remote patient monitoring systems as well as other applications associated with telemedicine. It is possible that the implementation of the adaptive transmission method in the transportation industry will improve the efficiency

as well as the dependability of the systems that are currently in place. For instance, the adaptive transmission strategy could be used in the process of putting in place communication systems for vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) interactions. These interactions involve the exchange of information between moving vehicles and fixed infrastructure. Applications for smart cities, such as intelligent traffic management systems, smart grids, and smart buildings, can be developed with the help of the adaptive transmission method, which can be utilized to design these applications. In this paper, we suggest a novel network architecture that blends cloud computing in order to lessen the load on cloud servers and the delay in the processing of tasks.

1.1 Contribution of the paper

In this paper, we will explore how cloud computing and the Internet of Things (IoT) might collaborate to speed up the processing of time-sensitive activities. We suggest a vehicle design that uses cloud computing to execute the desired function in order to reach this goal. We then go on to talk about how important it is to set up adaptive work offloading mechanisms and how quickly this needs to be done. Improving the completion rate of time-sensitive procedures was the driving force for the creation of the proposed offloading paradigm. This was the primary motivation behind creating the model. Adaptive methods for task offloading and transmission are employed in light of this consideration. Since the needs and available resources are subject to constant change, this algorithm dynamically organizes all of the jobs into various cloud link lists on the cloud. After that, the tasks in each list are cooperatively assigned among several nodes, taking into account their individual features before doing so. Shortly after the simulation model was presented, we conducted an experimental investigation to determine the model's efficacy.

2 Methods/experimental

Because of the increasing number of people who use the Internet of Vehicles (IoV), it is necessary to build a reliable data center that is in a position to provide support for the application services that are associated with the IoV. The provision of services for the Internet of Vehicles [12] is now regarded as the most widely deployed application of cloud computing technology. The term "cloud computing" refers to a kind of online data storage and processing in which customers have constant, anywhere-in-the-world access to shared computing resources on a pay-as-you-go basis. Cloud computing is also defined by the on-demand delivery of computing resources for a fee. Over the course of the past few years, IoV has increasingly relied on cloud computing and its accompanying technologies for storing, processing, and analyzing data. While this is going on, some Internet of Things applications are being relocated to data centers that are used for cloud computing in order to provide customers with connected services [13, 14]. On the other hand, the strain that is imposed on cloud servers in IoV grows as a consequence of the precipitous growth in the number of mobile terminals, such as vehicles. This rise in the number of mobile terminals has caused an increase in the amount of pressure placed on cloud servers. In addition, the fact that data centers for cloud computing are located in remote locations makes for longer delays in the processing of service requests. This is a critical problem that has to be addressed in IoV for latency-sensitive applications

[15, 16]. For instance, the ambulance has to receive information about the surrounding traffic conditions in real time in order to aid the driver so that it can get to the rescue location in a timely manner in the event that there is heavy traffic. In addition, in order to guarantee that people are able to drive their automobiles safely, it is necessary to have information about potential collisions that is updated in real time.

Software-defined networking (SDN) is now one of the most actively researched topics in the information technology industry [17, 18]. SDN is defined by its ability to operate networks in a methodical, centralized, and programmable way via the uncoupling of data planes and control planes. This is the defining property of SDN. Because of this characteristic, SDN is an essential technology for finding solutions to the challenges posed by the challenging growth and control of IoV infrastructures. Because of something called fog computing, it is now possible to bring the processing capacity of the cloud to the edge of the network. This is accomplished by the provision of computing, storage, and network services between the terminal devices and the cloud data centers. The phrase "fog computing" refers to a cloud that is physically placed closer to the end users and provides computing and associated services that have a reduced latency. Fog computing is also known as "edge computing." On the other hand, fog computing networks often consist of a number of pieces of network equipment that have a constrained capacity for computer processing. There is a possibility that a single piece of fog equipment may struggle to effectively assess substantial amounts of data [18, 19]. As a consequence of this, it is absolutely necessary to set up a fog computing network in order to carry out distributed computing by using a variety of various pieces of fog equipment. Utilizing load balancing techniques is another important step that must be taken to ensure that network loads are evenly distributed and that latency is kept to a minimum.

As a natural progression from the concept of fog computing, a number of academics have begun investigating the architecture of fog computing. They accomplished this by combining the design of fog computing with certain previous network topologies and making use of the advantages of fog computing to compensate for the inadequacies of the prevailing network designs. They achieved this by integrating the design of fog computing with certain preexisting network topologies. Lin and Shen [18] proposed a more condensed design for fog computing and also showed the application architecture of fog computing in smart grids. Lin and Shen [18] also discussed how fog computing may be used. A method for the optimization of task scheduling that is based on the Internet of Things (IoT) was proposed in [19, 20]. This approach takes into account both the time constraints and the financial implications of the scientific process for cloud computing. It is possible to successfully prepare for and manage resources in real time according to the specific circumstances of each scenario. The use of computation that takes place inside a network is a valuable addition to cloud computing. The cloud server is in charge of performing time-consuming calculations about the current status of the game and communicating any new information to the cloud nodes. Experiments have shown that this architecture is better than cloud computing and local clouds in terms of its capability to cut down on the length of time it takes for a game to reply as well as the amount of bandwidth that it needs [21].

A number of academics have simulated the fog network and produced systems that are capable of being optimized so that they may carry out in-depth studies on the fog

network. As an alternative to typical embedded systems, which have limits in terms of capacity, flexibility, and scalability, a software-defined embedded system that is enabled by fog computing was introduced in [22]. This system is a result of the convergence of software-defined networking and fog computing. A modeling study of the amount of time that it takes a task to execute is performed by the system. This analysis takes into consideration the amount of time that is spent on computing, I/O, and transmission. As a direct consequence of this, a low-complexity, three-stage approach to shortening the amount of time necessary for a service to reply has been devised. In addition, the consumption of energy, which is seen as an essential component of fog networks, is now the subject of a significant amount of focus and investigation. A conceptual model of the fog computing architecture was built in the work alluded to in [23], which also undertook mathematical modeling studies on service latency and energy consumption.

The authors of [24] have developed a delay-tolerant data transmission mechanism for the Internet of Things (IoV) of cloud computing systems. In the context of the Internet of Things application scenario, they investigated the differences between the standard cloud computing architecture in terms of the amount of energy used and the latency of the services. It has been shown that cloud computing gives advantages over cloud computing in terms of decreasing service latency while also lowering the amount of energy that is utilized. These benefits may be found in cloud computing. An examination and modeling of the delay and energy consumption of the subsystem, as well as an investigation into the trade-off between delay and energy usage [25], were carried out on the cloud-based architecture. The issue of load distribution that satisfies the optimization purpose of minimizing system energy consumption is evaluated while under the restrictions of delay constraints, and the associated optimization problem is handled by applying corresponding algorithms in order to find a solution.

In spite of the significant amount of effort that has been put into researching data transmission and data processing in IoV, there are still issues that need to be addressed [26–28]. The transmission over the existing network may be categorized as either delay-sensitive or delay-tolerant, depending on how sensitive or tolerant the data transmission is to delays. When transmitting data that can tolerate some amount of delay, the performance of the data transmission is not adversely affected by an increase in the amount of delay that is being sent [29–31]. However, if, during transmission, delay-sensitive data and delay-tolerant data are not differentiated from one another, it will definitely result in an increase in the demand placed on network resources as well as a waste of those resources. On the other hand, the mode of operation for conventional data processing is often cloud computing. On the other hand, the installation of servers for data storage and cloud computing is often a ways off. The transmission of data often results in an excessive burden on the network as well as an unneeded waste of network resources.

The use of adaptive transmission methods can have a variety of implications for users' privacy and security. An increase in the quantity of traffic that a device generates can make it more apparent to potential attackers; however, this can also put the device at risk of being attacked. Attackers may find it simpler to carry out some types of attacks, such as those involving a denial-of-service or a man-in-the-middle scenario, as a result of this. The adaptive transmission method may make it more difficult for consumers to exert control over the manner in which their data is sent,

resulting in reduced control over the data flows. This is due to the fact that the adaptive transmission method is able to dynamically alter the routing of data packets depending on the characteristics of the network. As a result, users may find it harder to prevent unauthorized parties from intercepting or watching their data. Because of the sophisticated nature of adaptive transmission methods, it might be difficult for users to comprehend how they operate and to recognize potential security flaws. As a result of this complexity, it may be simpler for attackers to take advantage of these vulnerabilities.

In recent years, in connection with the rise of cloud computing, researchers have begun studying load balancing strategies that are employed by cloud computing. A variety of academics have carried out this research. However, fog computing more closely resembles a cloud that is low to the ground, despite the fact that it is sometimes referred to as a supplement to cloud computing. Because of the heterogeneity of the fog computing network, the load balancing strategy that was established for cloud computing is not immediately relevant to fog computing. In spite of the fact that the data center for cloud computing is often situated in remote areas, it is still feasible to make use of the cloud data center as a distributed computing node in order to carry out a variety of tasks. This node has a great processing capacity but a significant transmission delay due to its location. As a consequence of these findings, a new network architecture for IoV that is based on cloud computing has been developed. This study investigates the delay optimization model while taking into account the limits imposed by the consumption of energy in cloud networks. Additionally, a technique for striking a balance between delay and resource utilization is presented.

3 Transmission framework of the proposed method

In this paper, the system model for adaptive transmission based on cloud computing in the IoV architecture is designed. The developing paradigms of cloud computing provide substantial support for offering efficient and effective services in today's world. Figure 1 illustrates the proposed cloud-based IoV architecture. Within this structure, there are many different base stations (BSs) and vehicle access points (VAPs) that are now awaiting assignments from their respective vehicles, etc. In addition to that, the cloud computing framework is presented here to manage the responsibilities that have been delegated to VAPs and BSs. The VAPs, which are represented by the notation $V_{\mathrm{AP}} = \left\{V_{\mathrm{AP}_1}, V_{\mathrm{AP}_2}, \ldots, V_{\mathrm{AP}_s}\right\}$ are used for the purpose of receiving task requests from vehicles by employing wireless signals, with s standing for the number of VAPs in the system.

The BSs, represented by the notation $\mathrm{BS} = \left\{B_1, b_2, \ldots, B_k\right\}$, are organized to deliver effective services in order to increase access speed and efficiency of service, with k standing for the magnitude of the differences across BSs.

Let us say there are N vehicles, and we want to represent them as $V = \{v_1, v_2, \ldots, v_N\}$ that need to delegate their computational task to the nodes.

Let us assume that every vehicle device has just one work that has to be done on the computer, and the task collection of V is represented by the notation $\mathrm{Task} = \mathrm{Task}_1, \mathrm{Task}_2, \ldots, \mathrm{Task}_N$.

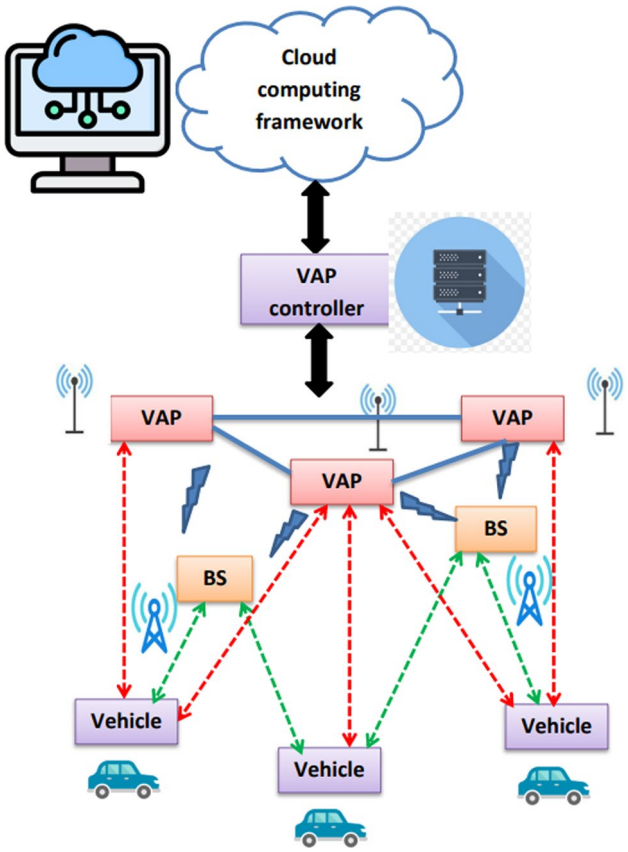

Fig. 1 Block diagram of cloud computing-based IOV architecture

3.1 Transmission target and offloading position

Let us assume that the starting location of transmission node is $l_{\text{start}}(i)$ in order to verifying the location of the offloading process. Here i denotes the node number. It is possible that each node $l_{\text{start}}(i)$ had two different types of edge nodes that are used for initial position selection. A decision about whether or not the vehicle v_N is within the umbrella of VAPs or BSs need to be taken into consideration. While delegating the work to the VAP that is closest to the located node $l_{\text{start}}(i)$, the VAPs should already exist to manage a huge number of individual tasks using vehicles that are located nearby. But these VAP s often have a low computational capacity. The maximum number of tasks that may be included inside them is denoted as Max(Task).

Let us consider the fact that the distances between the position of the node A and the vehicle v_n are denoted by the notation $d(A \rightarrow v_n)$. If the nearest VAP does not have enough space for hosting the nodes in the cloud network, then the node A will transport the data immediately for processing to the BS that is geographically closest to the location. When compared to the VAP, the BS provides a more extensive range of coverage options. If the vehicle is not included in the coverage supplied by any of the nodes in the cloud, node A is being processed at a place away from the original site.

$T(\text{BS} \leftrightarrow \text{VAP})$ is the metric that is used to decide which mode of transmission, VAP or BS, will be used. In the event that it is zero, the node will be offloaded to VAP, and in the event that it is one, the node will be offloaded to BS. This might also be stated as

$$T(\text{BS} \leftrightarrow \text{VAP}) = \left\{ \begin{array}{ll} 0, & \text{if } d(A \rightarrow v_n) < \text{Max(Task)} \\ 1, & \text{otherwise} \end{array} \right\}. \tag{1}$$

3.2 Transmission delay model

The duration of time it takes for data to be transmitted from one node in a cloud network to another has been taken into consideration in this article. We take into account the total amount of time that it takes for node A to relocate from the vehicle to its starting point. The formula for calculating the migration time for sending data from v_n to $l_{\text{start}}(i)$ is as follows:

$$\text{Time}(v_n \rightarrow l_{\text{start}}(i)) = \left\{ \begin{array}{ll} \frac{d(v_n \rightarrow l_{\text{start}}(i))}{\alpha_{\text{BS}}}, & \text{if } T(\text{BS} \leftrightarrow \text{VAP}) = 1 \\ \frac{d(v_n \rightarrow l_{\text{start}}(i))}{\alpha_{\text{VAP}}}, & \text{otherwise} \end{array} \right. . \tag{2}$$

Here, α_{BS} and α_{VAP} indicate the transmission rates in the coverage of BSs and VAPs, respectively.

Next, we consider the amount of time it takes for node A to propagate from the $l_{\text{start}}(i)$ to v_n. It is calculated as,

$$\text{Time}(A \rightarrow l_{\text{start}}(i)) = \left\{ \begin{array}{ll} \frac{d(A \rightarrow l_{\text{start}}(i))}{\alpha_{\text{BS}}} \cdot \omega, & \text{if } T(\text{BS} \leftrightarrow \text{VAP}) = 1 \\ \frac{d(A \rightarrow l_{\text{start}}(i))}{\alpha_{\text{VAP}}} \cdot \omega, & \text{otherwise} \end{array} \right. . \tag{3}$$

Here, ω is the metric between $l_{\text{start}}(i)$ and v_n.

The factors that are considered are as follows:

- The length of time required to wait for transmission.
- The length of time required to actually carry out transmission.

The paper applies the theory of queuing to the problem of determining how much time tasks in nodes have to wait before they can be processed. Because of this, not only will task queuing on cloud devices have greater practical significance, but also the theoretical underpinnings for it will be strengthened.

Let us assume that the arrival rate and the service rate of tasks in VAPs and BSs to be R_{arr} and R_{ser}, respectively. The service strengths of the VAP (S_{VAP}) and the BS (S_{BS}) is given by

$$S_{\text{VAP}} = \frac{R_{\text{arr}}}{P_{\text{VAP}} \cdot P_{\text{BS}}} \tag{4a}$$

$$S_{\text{BS}} = \frac{R_{\text{ser}}}{P_{\text{VAP}} \cdot P_{\text{BS}}} \tag{4b}$$

are the equations that determine the service strengths, respectively. P_{VAP} and P_{BS} stand for the computational power of the VAP and BS, respectively.

$$T_{\text{wait}} = \frac{\left(\frac{L(A)}{R_{\text{arr}}} + \frac{L(v_n)}{R_{\text{ser}}}\right)}{2} \tag{5}$$

$L(A)$ and $L(v_n)$ denote the length of data packet at node A and v_n. The overall computational delay (τ_{comp}) is

$$\tau_{\text{comp}}(l_{\text{start}}(i) \rightarrow v_n) = \text{Time}(A \rightarrow v_n) + \text{Time}(l_{\text{start}}(i) \rightarrow A) + T_{\text{wait}} \tag{6}$$

3.3 Cloud-based offloading: a computational delay-driven model

We have considered the computational delay for transmission in cloud network. The performance of cloud-based offloading involves the transmission of large amounts of data.

In addition, we suggest using a computational delay-driven model. Given a Task_N, the computing delay of offloading to vehicle v_n $\tau_{\text{comp}}(l_{\text{start}}(i) \rightarrow v_n)$ is determined by the task's start node $l_{\text{start}}(i)$ and vehicle node v_n. When Task_N is submitted, it is possible to compare the task delays from the two nodes in the cloud network. It is more likely to be offloaded through the cloud when $\tau_{\text{comp}}\ (v_n \rightarrow \text{cloud(node)}) = 0$. This is because the waiting time involved in cloud-based offloading may be ignored. It may be stated that the computational capacities of the cloud node and the vehicle node are dependent on the transmission rates of the core network and the cellular network. In addition, it is assumed that the input data size should not exceed the amount of storage space that the cloud has available.

Four different kinds of cloud links based on offloading scheduling are built, which are as follows:

1. Cloud-link (CL): Each and every activity on this list needs to be finished while connected to the cloud.
2. Vehicle-cloud-mixed link (VCM): The tasks on this list can either be offloaded to the car or to the cloud. Both options are available to the user.
3. Vehicle-mixed link (VM): The activities that are on this list have the potential to be transferred to the vehicle.
4. The cloud and the car are both viable options for offloading the tasks found in the vehicle-recommended link; however, it is recommended that the tasks be offloaded to the vehicle when the amount of time spent waiting in the vehicle is minimal.

In addition, in order to accomplish the production of the four connections, we provide a vehicle-driven division strategy. This strategy not only takes into account the resources that are necessary for the jobs, but it also takes into consideration the resources that are reachable from the various nodes. For the sake of clarity, note that whenever a vehicle is within the radio coverage of the cloud node, the cloud node converts into the potential offloading node for the vehicle. This is important information to keep in mind. The amount of time it takes to transmit the data should not be greater than the period of the vehicle's stay or the deadline for the assignment, and the amount of data should not be larger than what the cloud network is able to

hold all at once at a single instance. In addition, the length of time it takes to send the data should not be longer than the amount of time that has been set aside for the work.

3.4 Model for resource utilization

Because resource utilization is an important parameter for ensuring the stability of the cloud devices, it is essential to maintain a low resource utilization rate in order to keep the cloud devices appropriately occupied. When creating the offloading strategy, the virtualized approach makes use of the utilization of devices or all devices (that is, VAPs and BSs).

The offloading variables $O(A \rightarrow v_n)$ and $O(B \rightarrow v_n)$ are used to determine whether nodes A, B are offloaded to v_n for execution.

$$O(A \rightarrow v_n) = \begin{cases} 1, & \text{if } A \text{ is off loaded to } v_n \\ 0, & \text{otherwise} \end{cases} \tag{7a}$$

$$O(B \rightarrow v_n) = \begin{cases} 1, & \text{if } B \text{ is off loaded to } v_n \\ 0, & \text{otherwise} \end{cases} \tag{7b}$$

For determining the total number of devices that are actively functioning in the cloud is as follows:

$$\eta = \sum_{i=1}^{N} O(A_i \rightarrow v_n) + O(B_i \rightarrow v_n) \tag{8}$$

The utilization of the resources of VAP and BS may be estimated as follows:

$$U_{\text{VAP}}(A \rightarrow v_n) = \frac{1}{P_{\text{VAP}}} \sum_{i=1}^{N} O(A_i \rightarrow v_n) \tag{9a}$$

$$U_{\text{BS}}(A \rightarrow v_n) = \frac{1}{P_{\text{BS}}} \sum_{i=1}^{N} O(A_i \rightarrow v_n) \tag{9b}$$

Therefore, the average resource utilization of the cloud IoV devices is as follows

$$U_{\text{avg}} = \frac{1}{\eta} \left(\sum_{i=1}^{N} U(A_i \rightarrow v_n) + U(B_i \rightarrow v_n) \right) \tag{10}$$

Our objective is to decrease the transmission delay as much as possible so that we can improve the overall execution performance of all of the vehicular activities in IoV. At the same time, we want to increase the number of resources, which is characterized by the highest possible value of U_{avg} and the lowest possible value of τ_{comp}. This combination will improve the overall execution performance of all the vehicular activities in IoV. It is vital to cut down on the number of different stops along the route in order to achieve a low data routing latency via the connected car network.

4 Results

The system architecture described in this paper served as the foundation for the construction of the simulation model. In the concerned scenario, one cloud server, two BSs, and three VAPs are simulated. The cloud nodes provide complete coverage and are randomly distributed. The range of values for the storage capacity of BSs is uniformly created to be between 200 and 300. In addition, we set the capacity for VAPs to store data between the ranges of 10 and 15. Each vehicle will produce Taks in each time slot if the default setting is not changed. For each task, the needed computational resources are randomly distributed. In addition, the transmission from VAPs is sent to the vehicle through the cloud node.

Within an IoV architecture that is based on cloud computing, the ratio of resource utilization to transmission rate is inversely proportional to one another. This indicates that when there is a greater demand placed on the resources, the transmission rate will drop, and vice versa. This is because the quantity of transmission bandwidth that is available diminishes in direct proportion to the number of resources that are being used. The reason for this is due to the fact that there is a proportional decline. Figure 2 plots resource utilization vs transmission rate for $\tau_{\text{comp}}\ (v_n \rightarrow \text{cloud(node)}) = 0$. Figure 3 plots resource utilization vs transmission rate for $\tau_{\text{comp}}\ (v_n \rightarrow \text{cloud(node)}) \neq 0$.

An IoV architecture built on cloud computing must make efficient use of its resources if it is to realize its goal of achieving zero computational latency. This is essential for accomplishing the objective. To do this, one might make use of effective methods for managing cloud resources, like load balancing and virtualization. Edge computing, which moves certain processing and data storage tasks off of the cloud and onto edge devices, makes it possible to achieve this goal as well. Nevertheless, there is a possibility that lowering the amount of resources used will also lower the transmission rate. This is because the utilization of fewer resources leads to a lesser amount of available bandwidth for transmission, and this is the reason why this is the case.

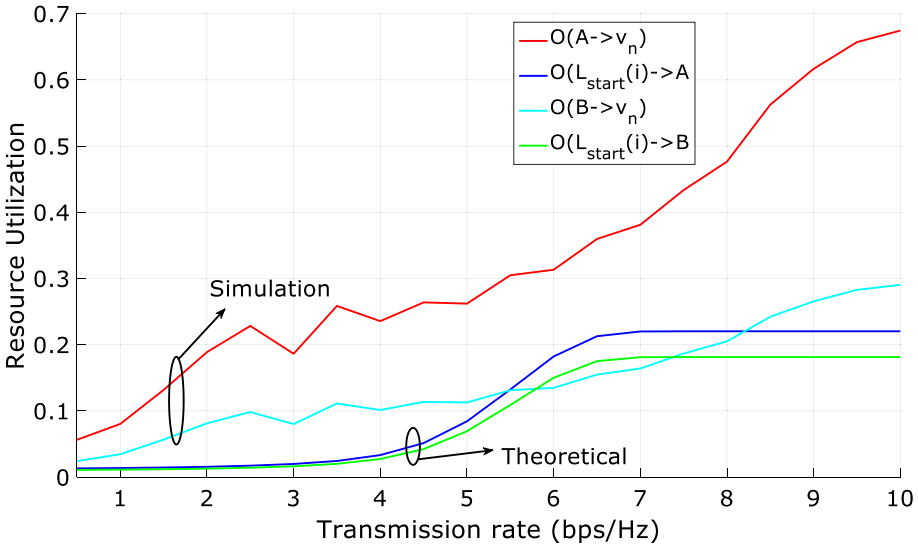

Fig. 2 Resource utilization vs transmission rate for $\tau_{\text{comp}}\ (v_n \rightarrow \text{cloud(node)}) = 0$

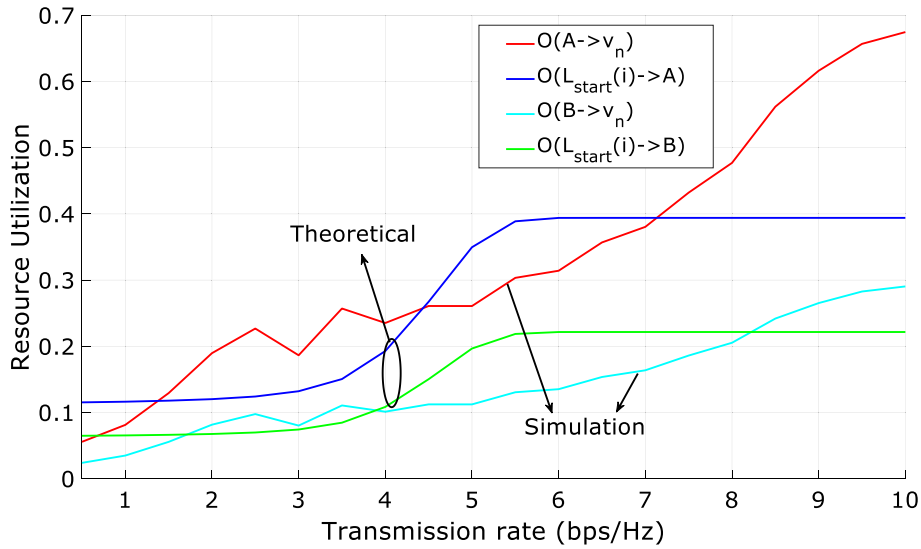

Fig. 3 Resource utilization vs transmission rate for τ_{comp} ($v_n \rightarrow$ cloud(node)) $\neq 0$

Therefore, with zero computational delay, it is essential to find a balance between the amount of resources being used and the pace at which they are being sent. Within an IoV architecture that is based on cloud computing, the following are some particular tactics that may be used to minimize resource utilization for zero computational delay. Employing effective ways for managing cloud resources may help to guarantee that resources are utilized effectively and that there is no needless waste of any kind. Using cloud architecture helps to balance the load between the cloud and the edge devices, which can also increase resource utilization and transmission rates. It can also help to balance the load between the cloud and the other devices.

The resource utilization and transmission rate in a cloud computing-based IoV architecture for zero or nonzero computational delay depend on the information that is being sent over the network, the computational resources that are accessible, and the total number of vehicles connected to the network. Figure 4 plots resource utilization vs transmission rate under varying number devices in the network for τ_{comp} ($v_n \rightarrow$ cloud(node)) $\neq 0$. Figure 5 plots resource utilization vs transmission rate under varying number devices in the network for τ_{comp} ($v_n \rightarrow$ cloud(node)) $= 0$. The resource utilization and transmission rate will both rise as the number of devices connected to the network grows. This is because there will be a rise in the quantity of data that needs to be sent and processed, which will require an increase in the number of resources that are available. There is no linear relationship between resource use and data transfer rate. For instance, there is still a probability that the transmission rate and the quantity of resources being utilized will not double, even if the total number of devices attached to the network doubles. This might be due to improved data processing efficiency on the part of the computer resources, or it could be the consequence of the network's architecture being able to manage the higher amount of traffic. These two scenarios are equally plausible.

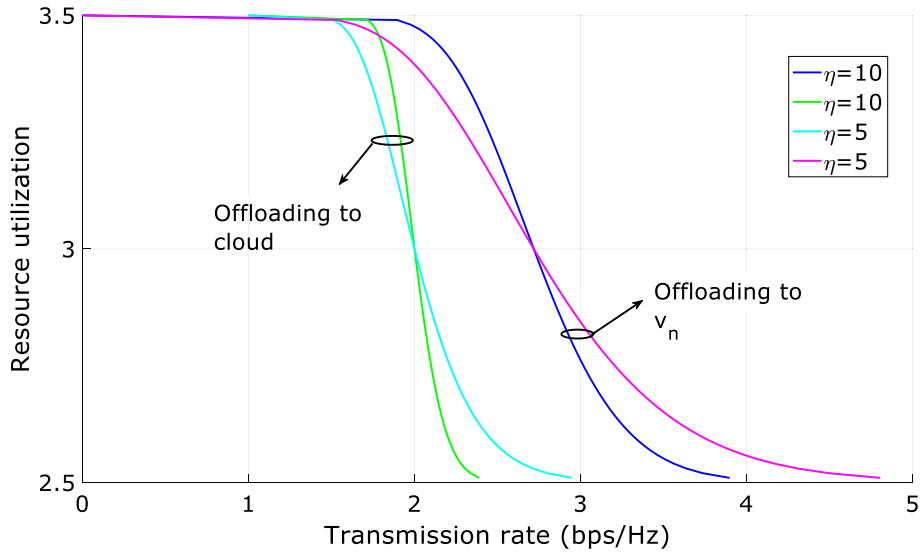

Fig. 4 Resource utilization vs transmission rate under varying number devices in the network for τ_{comp} ($v_n \rightarrow$ cloud(node)) $\neq 0$

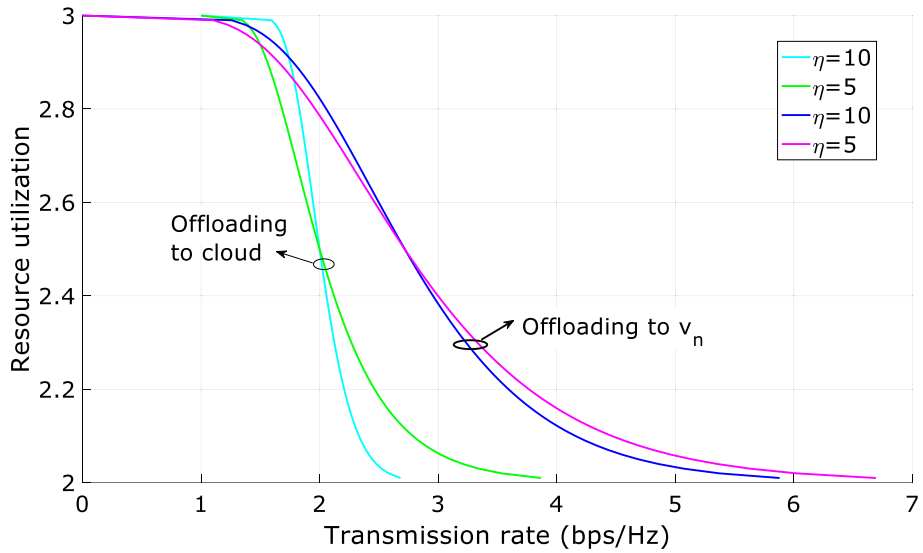

Fig. 5 Resource utilization vs transmission rate under varying number devices in the network for τ_{comp} ($v_n \rightarrow$ cloud(node)) $= 0$

If there is no error during the computation process, then the total amount of time that is spent making use of the available resources will need to be significantly increased in order to fulfill the prerequisites. This is because the required computing resources will need to be utilized in order to guarantee that the data can be handled in real time. The existence of this issue is a direct consequence of it. In an IoV architecture that is built on cloud computing, it is still feasible to maximize the efficiency of resource consumption while maintaining a high data transfer rate, even when the total number of connected devices may fluctuate from time to time. This is the case, even though it is possible. Not only the number of resources that are being used but

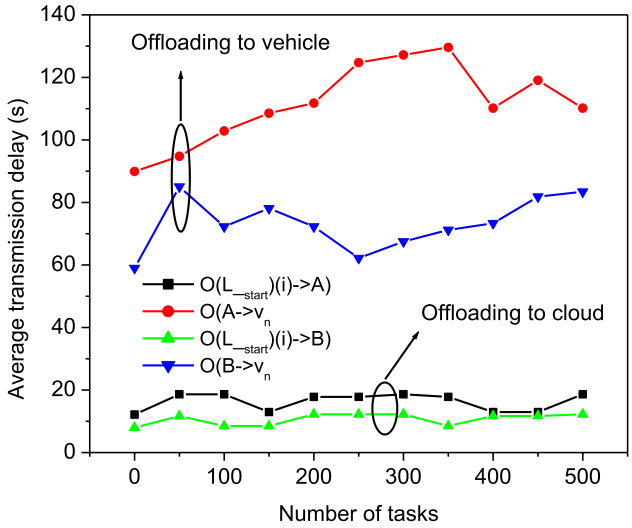

Fig. 6 Average transmission delay under varying number of tasks for τ_{comp} ($v_n \rightarrow$ cloud(node)) = 0

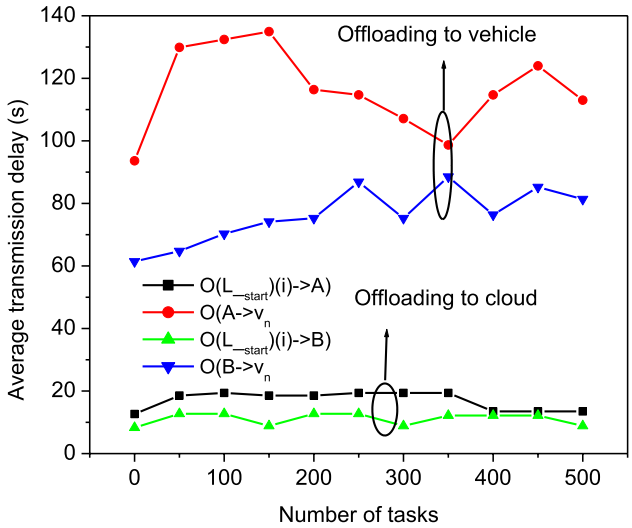

Fig. 7 Average transmission delay under varying number of tasks for τ_{comp} ($v_n \rightarrow$ cloud(node)) ≠ 0

also the rate at which data is being communicated can be significantly impacted by the kind of data that is being sent over the network. If a system only has a limited quantity of processing resources at its disposal, then it is quite likely that it will not be able to handle the data in real time.

The number of tasks that are being carried out will result in an increase in the typical amount of latency experienced by transmissions. This is because there will be a larger quantity of data to be conveyed and processed, both of which will require more time to complete. As a result, this will take longer. The average transmission latency is shown as a function of the number of tasks in Fig. 6, which has a value zero and is denoted by τ_{comp} ($v_n \rightarrow$ cloud(node)) = 0. Figure 7 plots average transmission delay under varying number of tasks for τ_{comp} ($v_n \rightarrow$ cloud(node)) ≠ 0. There may not

be a straight relationship between the amount of work being done and the typical amount of time lost in the gearbox. It is not always the case, for instance, that doubling the number of tasks would result in an equivalent increase in average transmission latency. This might be due to improved data processing efficiency on the part of the computer resources, or it could be the consequence of the network's architecture being able to manage the higher amount of traffic.

5 Discussion

If there is no requirement for processing to occur in close approximation to real time, then the quantity of jobs and the typical latency in transmission may not be as tightly related as they otherwise would be. This would be the case if there were a requirement for the processing to occur in real time. It is possible that this impact will have both positive and negative impacts. If it is necessary for there to be no latency in the computational processing, then the typical delay that takes place during transmission will be significantly lengthened. This is because the exploitation of computing resources will be required to ensure that the data can be processed in real time. The reason for this is due to the fact that the utilization of computational resources will be required. Because of this, there is a difficulty right now. Whether the computational delay is zero or not, it is difficult to offer an accurate evaluation of the precise link between the number of tasks and the average transmission delay in an IoV architecture that is based on cloud computing and has varying numbers of tasks. This is because the number of tasks can vary.

This holds true whether or not there is a computational delay. This is because the workload may change from one day to the next. However, by paying close attention to the numerous components of an IoV design that may be used to improve the average transmission delay, a reasonable approximation can be achieved. The average transmission delay may be minimized by paying careful attention to the numerous facets of an IoV design.

6 Conclusions

Recently, there have been a lot of marketing efforts aimed at the Internet of Vehicles (IoV) via vehicle access points (VAPs). Servers located in faraway clouds often handle routine computer processes related to the Internet of Things (IoT). Thus, the response time of tasks is significantly lengthened. Located near base stations (BSs) and virtual access points (VAPs), cloud servers provide several hosting alternatives for applications with diverse needs. Still, with all the BSs and VAPs crammed into this design, it might be hard to tell where computing activity in IoV go to dump. This is due to the fact that the architecture is very complex. This led to the development of a cloud computing infrastructure optimized for the Internet of Vehicles, as well as an adaptable method for computational offloading and transmission. This approach's goal is to maximize the cloud-based system's efficiency in terms of resource utilization, offloading, and transmission latencies for the activities it performs.

As part of this effort, we switch from testing with hardware-in-the-loop to testing with small-scale realistic IoV scenarios. Because of this change, the testing will be more precise. New to this, the research will use the previously stated framework to conduct investigations into new time-sensitive applications in IoV. Implementing adaptive

transmission control in the future can be facilitated by utilizing various machine learning techniques. In this case, the method of reinforcement learning could be applied to help create adaptive transmission mechanisms that can learn to perform optimally in different network environments. It is feasible that these strategies can be taught to achieve the highest level of operational efficiency.

Abbreviations

IoV	Internet of Vehicles
VAPs	Vehicle access points
QoE	Quality of experience
V2X	Vehicle-to-everything
BSs	Base stations
SDN	Software-defined networking
VM	Vehicle-mixed
CL	Cloud-link
VCM	Vehicle-cloud-mixed

Author contributions

BL and VV Li conceived of the study, and participated in its design and coordination and helped to draft the manuscript, ML, JL, and JQY, participated in the design of the study and performed the statistical analysis, BL Modified the paper.

Funding

This research was funded by Guangdong College Students' Science and Technology Innovation Cultivation Special Fund (Grant no. pdjh2023a0983), Guangdong Provincial Key Construction Discipline Scientific Research Capacity Enhancement Project (Grant no. 2021ZDJS136), Guangdong Province's Universities First-Class Professional Construction Project (Grant no. HS2022ZYJS05), Fundamental Research Funds of the Guangzhou Huashang College (Grant no. HSZB0202), Ministry of Education Supply and Demand Matching Employment Nurturing Internship Base Project (Grant no. 20230112031), Ministry of Education International Training Base for Chinese and Foreign Humanities Exchange(Grant no. CCIPE-WLJD-2022110020), Philosophy and Social Science Foundation of Guangdong (Grant no. GD20XXW06)..

Availability of data and materials

Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study.

Declarations

Competing interests

The author declares that there is nothing to declare.

Received: 22 November 2023 Accepted: 11 March 2024
Published online: 27 March 2024

References

1. J.E. Siegel, D.C. Erb, S.E. Sarma, A survey of the connected vehicle landscape—architectures, enabling technologies, applications, and development areas. IEEE Trans. Intell. Transp. Syst. **19**(8), 2391–2406 (2018)
2. Z. Tian, X. Gao, S. Su, J. Qiu, Vcash: a novel reputation framework for identifying denial of traffic service in Internet of connected vehicles (2019). Available: http://arxiv.org/abs/1902.03994
3. M. Amadeo, C. Campolo, A. Molinaro, Information-centric networking for connected vehicles: a survey and future perspectives. IEEE Commun. Mag. **54**(2), 98–104 (2016)
4. J.A. Guerrero-Ibanez, S. Zeadally, J. Contreras-Castillo, Integration challenges of intelligent transportation systems with connected vehicle, cloud computing, and Internet of Things technologies. IEEE Wirel. Commun. **22**(6), 122–128 (2015)
5. B. Jo, M.J. Piran, D. Lee, D.Y. Suh, Efficient computation offloading in mobile cloud computing for video streaming over 5G. Comput. Mater. Contin. **61**(2), 439–463 (2019)
6. C. Zhao, T. Wang, A. Yang, A heterogeneous virtual machines resource allocation scheme in slices architecture of 5G edge datacenter. Comput. Mater. Contin. **61**(1), 423–437 (2019)
7. J. Chen et al., Service-oriented dynamic connection management for software-defined Internet of Vehicles. IEEE Trans. Intell. Transp. Syst. **18**(10), 2826–2837 (2017)
8. A. Botta, W. de Donato, V. Persico, A. Pescapé, Integration of cloud computing and Internet of Things: a survey. Future Gener. Comput. Syst. **56**, 684–700 (2016)
9. J. Pan, J. McElhannon, Future edge cloud and edge computing for Internet of Things applications. IEEE Internet Things J. **5**(1), 439–449 (2018)
10. G. Premsankar, M. Di Francesco, T. Taleb, Edge computing for the Internet of Things: a case study. IEEE Internet Things J. **5**(2), 1275–1284 (2018)

11. Y. Wei, Z. Wang, D. Guo, F.R. Yu, Deep Q-learning based computation offloading strategy for mobile edge computing. Comput. Mater. Contin. **59**(1), 89–104 (2019)
12. H. Gao, Y. Duan, L. Shao, X. Sun, Transformation-based processing of typed resources for multimedia sources in the IoT environment. Wirel. Netw. (2019). https://doi.org/10.1007/s11276-019-02200-6
13. H. Gao, W. Huang, X. Yang, Applying probabilistic model checking to path planning in an intelligent transportation system using mobility trajectories and their statistical data. Intell. Autom. Soft Comput. **25**(3), 547–559 (2019)
14. X. Wang, Y. Ting-Ting, H. Shuang-Shuang, Parallel Internet of Vehicles: the ACP-based networked management and control for intelligent vehicles. Acta Autom. Sin. **44**(8), 1391–1404 (2018)
15. P.-W. Tsai, C.-W. Tsai, C.-W. Hsu, C.-S. Yang, Network monitoring in software-defined networking: a review. IEEE Syst. J. **12**(4), 3958–3969 (2018)
16. O. Lemeshko, O. Yeremenko, Enhanced method of fast re-routing with load balancing in software-defined networks. J. Electr. Eng. **68**(6), 444–454 (2017)
17. Q. Wang, S. Guo, J. Liu, Y. Yang, Energy-efficient computation offloading and resource allocation in fog computing for Internet of every-thing. China Commun. **16**(3), 32–41 (2019)
18. Y. Lin, H. Shen, Cloud fog: towards high quality of experience in cloud gaming, in *Proceedings of the 44th International Conference Parallel Process* (Beijing, 2015), pp. 500–509.
19. X. Ma, H. Gao, H. Xu, M. Bian, An IoT-based task scheduling optimization scheme considering the deadline and cost-aware scientific workflow for cloud computing. EURASIP J. Wirel. Commun. Netw. **2019**(1), 249 (2019). https://doi.org/10.1186/s13638-019-1557-3
20. N.B. Truong, G.M. Lee, Y. Ghamri-Doudane, Software defined networking-based vehicular ad hoc network with fog computing, in *Proceedings of the IEEE International Symposium on Integrated Network Management* (Ottawa, 2015), pp. 1202–1207. https://doi.org/10.1109/INM.2015.7140467.
21. K. Intharawijitr, K. Iida, H. Koga, Analysis of fog model considering computing and communication latency in 5G cellular networks, in *Proceeding of the IEEE International Conference on Pervasive Computing and Communications Workshops* (2016), pp. 1–4. https://doi.org/10.1109/PERCOMW.2016.7457059
22. D. Zeng, L. Gu, S. Guo, Z. Cheng, S. Yu, Joint optimization of task scheduling and image placement in fog computing supported software-defined embedded system. IEEE Trans. Comput. **65**(12), 3702–3712 (2016)
23. S. Sarkar, S. Misra, Theoretical modelling of fog computing: A green computing paradigm to support IoT applications. IET Netw. **5**(2), 23–29 (2016)
24. R. Deng, R. Lu, C. Lai, Optimal workload allocation in fog-cloud computing toward balanced delay and power consumption. IEEE Internet Things J. **3**(6), 1171–1181 (2016)
25. H. Gao, Y. Xu, Y. Yin, W. Zhang, R. Li, X. Wang, Context-aware QoS prediction with neural collaborative filtering for Internet-of-Things services. IEEE Internet Things J. (2019). https://doi.org/10.1109/JIOT.2019.2956827
26. S.-M. Oh, J. Shin, An efficient small data transmission scheme in the 3GPP NB-IoT system. IEEE Commun. Lett. **21**(3), 660–663 (2017)
27. Q. Wang, S. Guo, J. Liu, Y. Yang, Energy-efficient computation offloading and resource allocation for delay-sensitive mobile edge computing. Sustain. Comput. Inform. Syst. **21**, 154–164 (2019)
28. H. Gao, W. Huang, Y. Duan, X. Yang, Q. Zou, Research on cost-driven services composition in an uncertain environment. J. Internet Technol. **20**(3), 755–769 (2019)
29. S. Yan, A. Aguado, Y. Ou, Multi-layer network analytics with SDN-based monitoring framework. IEEE/OSA J. Opt. Commun. Netw. **9**(2), 271–279 (2017)
30. L. Tang, R. Liang, Y. Zhang, Load balance algorithm based on POMDP load-aware in heterogeneous dense cellular networks. J. Electron. Inf. Technol. **39**(9), 2134–2140 (2017)
31. Y.F. Zhou, N. Chen, The LAP under facility disruptions during early post-earthquake rescue using PSO-GA hybrid algorithm. Fresenius Environ. Bull. **28**(12A), 9906–9914 (2019)

Publisher's Note