ISSN Online: 2821-1693

Transactions on Machine Intelligence

Journal Homepage: https://tmachineintelligence.ir/

Resource Management in Vehicular Fog Networks Based on Contract Theory

R. Haghighizadeh[1], Z. Zali[2,*]

[1] Department of Electrical Engineering, School of Electrical and Computer Engineering, Isfahan University of Technology, Isfahan, Iran

[2] Assistant Professor, Department of Electrical Engineering, School of Electrical and Computer Engineering, Isfahan University of Technology, Isfahan, Iran

ARTICLE INFO

Article History:
Received 22 November 2024
Received in revised form 5 February 2025
Accepted 27 May 2025
Available online 1 June 2025

Keywords:
Fog Computing, Task Allocation, Quality of Service, Vehicular Fog Computing

ABSTRACT

Fog computing is a distributed infrastructure that extends computing, communication, and storage capabilities toward the network edge. Compared to cloud computing, fog computing can support delay-sensitive service requests while reducing energy consumption and traffic congestion. Fog computing contributes to efficient resource utilization and improved performance in terms of latency, bandwidth, and energy consumption. However, one of the major challenges in fog computing is the limited computational capacity of fog nodes under increasing daily demand, especially during peak hours, which can lead to severe performance degradation. Therefore, an optimal mechanism is required to ensure satisfactory quality of service (QoS). Integrating fog computing with vehicular ad hoc networks, leading to vehicular fog computing (VFC), has emerged as a promising solution to reduce overload at base stations and minimize processing delays during peak periods. In this approach, the surplus computational resources of nearby vehicles are utilized as an on-demand, low-cost option. Consequently, the computing resources provided by a large group of vehicles can be aggregated to alleviate network congestion during peak hours without additional servers, enabling real-time computational scalability. Nevertheless, the large-scale deployment of vehicular fog networks still faces several critical challenges, such as the lack of efficient incentive mechanisms and task assignment strategies. In this study, we first propose a solution to minimize network delay from the perspective of integrated contract-based optimization. Next, the problem of computational task allocation is formulated as a two-sided matching problem between vehicles and users, and a stable, QoS-aware matching algorithm is introduced to solve it. Finally, task offloading decisions are performed to minimize total network latency. The proposed scheme can effectively guarantee network load balancing and improve the utilization of idle vehicular resources.

[*] Corresponding Author: zali@iut.ac.ir
Assistant Professor, Department of Electrical Engineering, School of Electrical and Computer Engineering, Isfahan University of Technology, Isfahan, Iran

http://dx.doi.org/10.22034/TMI.2025.244841

1. INTRODUCTION

Emerging applications in modern vehicles, such as autonomous driving, video streaming, and online gaming, generate a large number of tasks that are both computationally intensive and delay-sensitive [1]. A common approach to handling these tasks is to utilize cloud computing [2]. Cloud computing is a computational paradigm that aims to provide users with configurable computing resources such as networks, servers, storage, and user services so that each user can conveniently access these resources from any location and at any time [3].

However, the long transmission distance between end devices and cloud servers introduces several challenges, including high latency [4], unstable connections, network congestion [5], and high bandwidth demand [6], particularly when both the speed and volume of generated data increase. These issues significantly affect the quality of service (QoS) [7] and the quality of experience (QoE) for users [8]. Moreover, in cloud computing, due to the massive volume of data, user data and information are first processed and stored over a specific time interval and only then delivered to users in a separate process, making it unsuitable for real-time applications [9].

To address this issue, edge computing has been proposed, which offloads task processing from the cloud to network edges to reduce transmission distances [10]. Nevertheless, deploying a large number of edge servers to ensure geographic coverage and maintain service quality inevitably results in high operational costs and the potential waste of substantial resources. In this context, fog computing has emerged as a promising solution [11].

Fog computing is a distributed infrastructure in which data, computation, storage, and applications are dispersed locally between data-generating devices and the cloud. Today, there is a transition from centralized cloud computing infrastructures to distributed fog computing infrastructures. Both fog and cloud computing provide storage space, applications, and data access for users; however, fog computing is closer to the end user and has a wider geographical distribution [12]. The devices within the fog are considered as nodes, and fog computing is often referred to as “edge computing” because it extends the cloud computing paradigm to the network edge. It acts as a platform capable of providing computing, storage, and networking services between end nodes and the cloud. In other words, fog computing hosts services at the network edge and on end devices, reducing service latency and improving QoS, which leads to an enhanced user experience [13].

Integrating fog computing with vehicular ad hoc networks, resulting in vehicular fog networks, allows for improved communication efficiency and addresses typical constraints such as unstable connection latency, network congestion, high bandwidth demand, location awareness, and real-time response, which are common in intelligent traffic control, driving safety applications, and entertainment services [14]. Moreover, local task processing at fog nodes preserves network resources and reduces bandwidth consumption, as it eliminates the need to transmit data to the cloud for analysis. These requirements are particularly critical in hostile environments, such as urban warfare or battlefields involving military vehicles [15].

In [16], a three-layer architecture comprising the cloud layer, micro-cloud layer, and fog layer is proposed for vehicular fog networks. Due to the dynamic nature of fog nodes in vehicular networks, clouds are necessary to enhance connectivity. To increase the capacity of the fog layer, both parked and moving vehicles are utilized to provide computing resources for data processing. The main objective is to minimize the number of micro-cloud servers while optimizing latency, prioritizing task processing at the fog layer, and maintaining load balance between cloud and fog nodes. Similarly, in [17], the same three-layer vehicular fog architecture is considered for energy optimization using a deep reinforcement learning algorithm, which satisfies user timing constraints and balances computational loads across fog nodes.

To incentivize vehicles to share their resources, a base station deploys a contract-based mechanism without precise knowledge, relying instead on statistical information such as average local computational load and probability distributions of fog vehicle types. The primary goal of this incentive mechanism is to reduce the load on the base station while increasing resources shared by fog nodes. In another study [15], a novel learning-based matching algorithm is proposed to minimize long-term task offloading delays in scenarios where multiple vehicles simultaneously select a fog server. This approach enables user vehicles to identify the optimal fog nodes based on local information. However, one of the main limitations of that study is the lack of consideration for fog server volatility. Given high mobility and complex road conditions, a user vehicle may face transient fog servers, where some existing servers suddenly disappear while new servers appear. Such fluctuations significantly impact the design of learning-based resource allocation.

In [18], mobility is leveraged to mitigate the limitations arising from intermittent connectivity, as mobility increases the likelihood of encounters between nodes, thereby improving connectivity and task offloading opportunities. Many vehicular fog network applications rely on consecutive V2I and V2V communications; thus, offloading latency is strongly related to the time intervals between communications. Vehicle mobility reduces these intervals, thereby decreasing offloading delay. In [19], to address the typically unpredictable mobility of users, a Lyapunov optimization approach is employed to decompose the overall optimization problem into a series of real-time optimization problems, which do not require prior knowledge such as user mobility. Initially, an approximate algorithm based on Markov approximation is designed to reach an optimal solution.

Another critical challenge is maintaining quality of service (QoS). In [20], a two-layer vehicular fog architecture is proposed, comprising cloud servers and both fixed and mobile fog servers, which increases the capacity of the fog layer. Users submit tasks with strict delay constraints that must be completed within specified deadlines, while servers in this network have heterogeneous computation, communication, and storage capabilities. Therefore, an adaptive task offloading algorithm is proposed that considers the heterogeneous capabilities of servers and the high mobility of vehicles, with the aim of optimizing task processing times within the required deadlines.

2. PROPOSED MODEL

As illustrated in [12] and shown in Figure 1, a vehicular fog network architecture is considered, comprising a base station, a set of fog-enabled vehicles, and pedestrians as users. The base station in each cell is responsible for coordinating intra-cell communication resources and allocating computational resources. During peak hours, when the load on the base station increases and it becomes unable to process all incoming computational tasks independently, a group of vehicles is employed to act as fog nodes to alleviate the overload by offloading user tasks. In this way, vehicles can serve as fog nodes by sharing their idle computational resources to process tasks.

Each user generates tasks that, due to limited computational resources, are intended to be processed either by the base station or by vehicular fog nodes, as depicted in Figure 1. With a properly designed incentive mechanism, each vehicle can actively adjust the number of resources it shares to maximize its own utility. Multiple users may exist within the same cell. Each user generates a set of computational tasks, each of which can be processed either by the base station or offloaded to a vehicular fog node [12].

For simplicity, a discrete-time model is adopted, in which time is divided into slots. The sets of vehicles and users within the coverage area of the base station remain fixed during each time slot and are denoted as

$$V_M = \{v_1, \dots, v_m, \dots, v_M\} \text{and} U_N = \{u_1, \dots, u_n, \dots, u_N\}$$

During each time slot, it is assumed that each user for example, the n-th user has a computational task that must be processed. The key characteristics of a computational task can be described by the triple $\{D_n, C_n, \tau_n\}$, where $D_n$denotes the data size of the computational task, $C_n$represents the computational resources required for processing, and $\tau_n$indicates the task's delay constraint.

The system model can be extended to scenarios in which a user has multiple computational tasks to be processed during a time slot. In such cases, multiple tasks can be aggregated and considered as a single task with a larger data volume and higher computational demand.

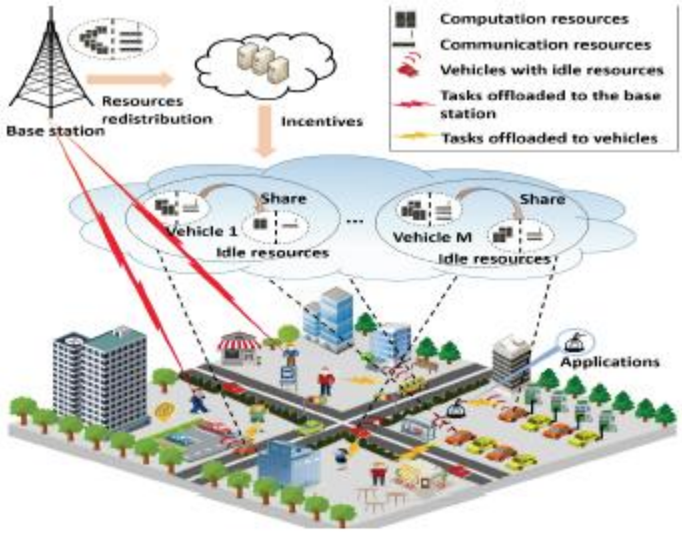


Fig. 1. Framework of the Vehicular Fog Network [2].

2.1. Contract Problem and Incentive Mechanism in Vehicular Fog Networks

The primary objective of the contract problem is to maximize the expected profit of the base station while ensuring that the rewards received by the fog nodes are non-negative. In the proposed approach, we first introduce the modeling of vehicle types, then define the utility functions for both the base station and the vehicles, and finally formulate and solve the optimization problem [12].

2.1.1. Determination of Vehicle Types

The priority of a vehicle for sharing its resources is defined by its type. A vehicle with a higher type, compared to a lower-type vehicle, is willing to share more of its resources and act as a fog node, thus enabling it to earn higher utility. Therefore, employing higher-type vehicles is essential for the base station.

Since the number of vehicles in a cell is typically limited, the set of vehicle types belongs to a finite and restricted space. The M vehicles in the set V_M can be sorted in ascending order according to their preferences and classified into K types. The set of vehicle types is denoted as $K = \{1, \ldots, k\}$, and $\Theta = \{\theta_1, \ldots, \theta_K\}$ represents the set of computational capabilities of each fog vehicle type available for resource sharing [12].

$$\theta_1 < \cdots < \theta_k < \cdots < \theta_K \qquad k \in K \tag{1}$$

Next, the type of each vehicle is determined. By sharing resources, the task processing delay for the fog nodes increases; however, this delay must remain below $\Delta\tau_{m,\max}$ as defined in Equation (2) to ensure both quality of service (QoS) and user experience.

$$\triangle \tau_m = \frac{c_m}{\delta_{m.0} - \delta_m} - \frac{c_m}{\delta_{m.0}} \leq \triangle \tau_{m.max} \tag{2}$$

Denotes the computational workload of local tasks that must be processed by vehicle V_m. $\delta_{m,0}$ and δ_m represent the total available computational resources and the resources that the vehicle shares, respectively. δ_m^{upper} indicates the maximum amount of resources that a type-m vehicle can share, which is defined in Equation (3):

$$\delta_m \leq \frac{\delta_{m.0}^2 \triangle \tau_{m.max}}{\delta_{m.0} \triangle \tau_{m.max} + c_m} = \delta_m^{upper} \tag{3}$$

It is assumed that δ_m^{upper}lies within the range $\{\delta_{\min}, \delta_{\max}\}$, representing the minimum and maximum values of δ_m^{upper}, respectively. Then, according to Equation (4), this range is divided into $\boldsymbol{K}$sub-ranges of equal length, and the lower bound of each sub-range is defined as θ_k[12].

$$\theta_k = \delta_{min} + \frac{k-1}{K}(\delta_{max} - \delta_{min}) \tag{4}$$

It can be inferred that $\theta_k$increases with $\delta_{m,0}$and decreases with C_m. This definition is consistent with practical conditions; for example, a vehicle with light local tasks and abundant idle resources can share more of its computational capacity. Consequently, it can achieve higher utility and has a stronger preference for resource sharing.

In an information-asymmetric scenario, the base station does not have precise knowledge of the type of each vehicle. Instead, only statistical knowledge of vehicle types, obtained through long-term measurements or historical observations, is available. It is assumed that the base station only knows that there are $k$vehicle types in total. Accordingly, the type of vehicle $V_m$is denoted as Kif [12]:

$$\theta_k \leq \delta_m^{upper} < \theta_{k+1} \tag{5}$$

2.1.2. Utility of the Base Station

The base station designs Kcontracts for K types of vehicles, meaning that one contract is considered for each vehicle type. For example, the contract assigned to a vehicle of type K is denoted as $\{\delta_k, \pi_k\}$, where δ_k represents the computational resources required, and π_k is the corresponding reward. The complete set of contracts is expressed as $C = \{\delta_k, \pi_k\}$.

Thus, by signing the contract $\{\delta_k, \pi_k\}$with a vehicle of type K, the total utility of the base station and users is obtained as follows:

$$U_S(\{\delta_k\}.\{\pi_k\}) = U_{BS}(\{\delta_k\}.\{\pi_k\}) + M\sum_{k=1}^{K} \lambda_k U_K^V(\delta_k.\pi_K) \tag{6}$$

2.1.3. *Formulation of the Optimization Problem in an Information-Asymmetric Scenario*

The objective of the base station in an information-asymmetric scenario is to maximize its profitability by optimizing each contract. Accordingly, the corresponding optimization problem can be formulated as follows:

$$\max_{(\{\delta_k\}.\{\pi_k\})} \quad U_{BS}(\{\delta_k\}.\{\pi_k\}) \tag{7}$$

$$\text{s.t} \quad \theta_k\pi_k - \delta_k \geq 0 \qquad \forall k \in K$$

$$\theta_k\pi_k - \delta_k > \theta_k\pi_{k'} - \delta_{k'} \qquad \forall k.k' \in K \tag{7a,b}$$

$$0 \leq \delta_1 < \cdots < \delta_k < \cdots < \delta_K \qquad \forall k \in K \tag{7c}$$

$$\delta_k \leq \theta_k \qquad \forall k \in K \tag{7d}$$

Constraint (7a) ensures that a vehicle of type $k$receives a non-negative reward if it selects the contract $\{\delta_k, \pi_k\}$. Constraint (7b) guarantees that a vehicle of type $k$achieves maximum utility when it chooses the contract $\{\delta_k, \pi_k\}$ designed for its own type. Constraint (7c) ensures that the amount of requested resources from a vehicle of type $k-1$ is less than that of a vehicle of type $k+1$. Constraint (7d) indicates that the shareable resources of the edge nodes must exceed the requested resources, so that the utilization does not exceed the nodes' capacity.

By examining the Hessian matrix, it can be concluded that the defined optimization problem is convex.

2.1.4. Formulation of the Optimization Problem in an Information-Symmetric Scenario

If the base station has complete knowledge of the type of each vehicle, it can design contracts according to the available resources of edge vehicles. This allows it to increase the amount of resources shared by the vehicles while also maximizing its own utility. The contracts must ensure that the utility of each vehicle is non-negative, and the base station can set the utility of vehicles to zero if needed. Accordingly, the optimization problem can be defined as follows:

$$\max_{(\{\delta_k\}.\{\pi_k\})} \quad U_{BS}(\{\delta_k\}.\{\pi_k\}) \tag{8}$$

$$\text{s.t} \qquad \theta_k\pi_k - \delta_1 = 0 \qquad \forall k \in K \tag{8a}$$

$$0 \le \delta_1 < \cdots < \delta_k < \cdots < \delta_K \qquad \forall k \in K \tag{8b}$$

$$\delta_k \le \theta_k \qquad\qquad \forall k \in K \tag{8c}$$

2.2. Task Allocation Problem Based on Matching with QoS Awareness

In this section, we address the task scheduling and allocation between users and vehicular edge nodes based on the proposed QoS-aware algorithm. The process involves several steps, including user prioritization, task determination, evaluation conditions, creation of the user priority list, and resolving competition among users.

Our goal is to minimize network overhead and latency while ensuring that the QoS requirements of tasks are met. Accordingly, we propose a QoS-aware resource allocation scheme that jointly considers the relationships between edge nodes and tasks, as well as data transmission and computational resource allocation, to optimize task offloading decisions and reduce latency.

In the proposed algorithm, since the tasks are delay-sensitive, the most important metric for QoS is latency. Tasks are categorized into four groups based on their delay tolerance:

$$\begin{cases} 0 < \tau_n \le 0.5s & \rightarrow Q1 \\ 0.5s < \tau_n \le 1s & \rightarrow Q2 \\ 1s < \tau_n \le 1.5s & \rightarrow Q3 \\ 1.5s < \tau_n \le 2s & \rightarrow Q4 \end{cases} \tag{9}$$

Thus, the total delay when the task of user n is assigned to vehicle V_m is the sum of the transmission delay and the task execution delay.

$$T_{n.m}^{total} = T_{n.m}^{t} + T_{n.m}^{c} \tag{10}$$

In the proposed model, the user-tolerable delay constraint, the comparison between the resources shared by edge nodes and the computational resources required by users, as well as the communication distance between users and edge nodes, are considered. The main objectives of evaluating these conditions in the network are to increase the matching speed between users and edge nodes, reduce the size of users' priority lists, and decrease network complexity.

According to Equation (11), if an edge node is outside the communication coverage radius of a user, it is removed from the user's priority list and the offloading request is rejected. This in turn reduces the size of the users' priority lists and lowers computational complexity.

$$d_{n.m} < d_n \tag{11}$$

In the above equation, $d_{n,m}$ denotes the distance between user n and vehicle m, while d_n represents the communication radius of user n.

Moreover, due to the high mobility of vehicles, a vehicle may move out of the user's communication range during data transmission, which may result in task offloading failure. Therefore, according to Equation (12), the residence time of a vehicle within the communication range of user $n$is given by:

$$\tau_{n.m}^{o} = \frac{\hat{d}_{n.m}}{\bar{v}_m} \tag{12}$$

$\bar{V}_m$ denotes the average speed of vehicle m, and $\hat{d}_{n,m}$ represents the boundary point of the user's communication radius in the direction of the vehicle's movement. According to the following condition, an offloading request is acceptable only if the transmission delay is shorter than the residence time of the vehicle within the communication range of the user:

$$T_{n.m}^{t} < \tau_{n.m}^{o} \tag{13}$$

2.3. Communication Model and Task Assignment Optimization

To minimize the total delay resulting from task offloading to vehicular fog nodes, an optimal fog node must be selected among the available nodes to execute each user task. The objective is to minimize the overall network latency. Therefore, the problem is formulated as an optimization model in which the objective function aims to minimize the total system delay caused by task offloading.

$$\min_{X} \quad \sum_{n=1}^{N} \sum_{m=1}^{M} x_{n.m} T_{n.m}^{total} \tag{14}$$

$$\text{s.t} \quad \sum_{v_m \in V_M} x_{n.m} \leq 1 \qquad \forall u_n \in U_N \tag{14a}$$

$$\sum_{u_n \in U_N} x_{n.m} \leq 1 \tag{14b}$$

$$T_{n.m}^{total} \leq \tau_n \qquad \forall U_n \in u_N . \forall V_m \in v_M \tag{14c}$$

$$T_{n.m}^{t} \leq \tau_{n.m}^{o} \qquad \forall U_n \in u_N . \forall V_m \in v_M \tag{14d}$$

To provide an appropriate solution, the optimization problem can be transformed based on its structure into a two-sided matching problem. The matching problem can be represented by the triplet $\{U_n, V_m, F_n\}$, where U_n and V_m denote two distinct and finite sets of participants, namely the users and the vehicles, respectively. F_n represents the preference list set of user n.

Specifically, each user aims to minimize its individual task delay under the defined system constraints. The matching $\phi(U_n)=V_m$ is therefore defined as a one-to-one mapping between users and vehicular fog nodes, determined according to the users' preference lists.

To implement the two-sided matching mechanism, each user constructs its own preference list by ranking the vehicles. For user U_n, different delay performance values can be obtained when it is temporarily paired with different vehicles. Accordingly, the preference of user U_n over vehicle V_m is computed using the following relation:

$$G_{n.m}|_{\phi(U_n)=v_m} = \frac{1}{T_{n.m}^{total}} \tag{15}$$

Accordingly, user U_n is said to prefer vehicle V_m over vehicle $V_{m'}$ if:

$$V_m \geq V_{m'} \rightarrow G_{n.m}|_{\phi(U_n)=v_m} > G_{n.m'}|_{\phi(U_n)=v_{m'}} \tag{16}$$

In the matching phase, when a fog vehicle receives more than one request from users, metrics such as delay tolerance and the Quality of Service (QoS) requirements of users determine the final matching outcome. The user with the higher priority is first matched with the candidate fog vehicle, and that vehicle is then removed from the preference lists of the remaining users. This mechanism leads to a reduction in the total network delay.

Furthermore, if competition arises among users with the same priority level, in order to utilize the shared fog resources more efficiently and ensure the profitability of the base station, the user requiring a higher amount of computational resources will be matched with the fog vehicle according to relation(17).

$$C_m < C_k \tag{17}$$

2.4. Algorithm Procedure

The QoS-aware matching process proposed for task allocation and establishing associations between users and vehicular fog nodes is summarized in Algorithm 1.

Algorithm 1. QoS-Aware Matching Algorithm

Input:
Set of users $U_N = \{u_1, \dots, u_n, \dots, u_N\}$,
Set of vehicular fog nodes $V_M = \{v_1, \dots, v_m, \dots, v_M\}$,
Matching parameters for each user initialized as $\phi = 0$,
Number of requests received by each vehicle denoted by X.
Output:

$\phi(U_n)$

1. Classify users into four groups according to their QoS level and delay tolerance τ_n.
2. Evaluate the feasibility conditions using (11), (12), and (13).
3. Construct the preference list of each user $F_n$based on (15) and (16).
4. **Repeat while** $\exists \phi(U_n) = 0$:
5. For each unmatched user:
6. Send a request to the highest-ranked vehicular fog node in its updated preference list.
7. For each vehicular fog node:
8. **If** $X \geq 1$, **then**
9. Matching among users with different QoS requirements is performed based on their QoS priority.
10. **Else**, perform matching according to (18) based on the computational resource demand of each user.
11. **If** $X = 1$, **then**
12. The match is directly established according to the received request.
13. **End if**

3. RESULTS

3.1. Contract Problem and Incentive Mechanism Design in Vehicular Fog Networks

To address the resource-sharing problem, an incentive mechanism based on contract theory is proposed to encourage vehicular fog nodes to share their computational resources for task offloading. The contract is designed according to the unique characteristics of each type of vehicle, with the objective of maximizing the expected profit of the base station by optimizing the contract items, including the requested computational resources and the corresponding rewards. The resulting optimization problem is convex and can be solved using the Karush-Kuhn-Tucker (KKT) conditions.

A set of simulations is carried out to examine the feasibility and effectiveness of the proposed contract-based incentive mechanism. We consider a single cell consisting of one base station, a group of vehicular fog nodes, and a set of users. The available sharable computing resources of vehicular fog nodes are assumed to lie within the range of 370–770 MB. The simulation parameters are summarized in Table 1. The proposed scheme is evaluated by comparing it with the contract mechanism under the complete-information scenario, which is considered as the performance benchmark.

Table 1. Simulation Parameters [12]

Parameter	Value
Number of vehicular fog nodes M	5–20
Number of users N	6–30
Data size of user tasks D_n	100–200 MB
Computational workload of tasks C_n	100–400 MB
Delay constraint τ_n	0–2 s
Vehicle speed V_m	2–50 m/s
Cell radius R	1000 m
User communication radius R_n	200 m
Base-station computing resources δ_{BS}	5 GB
User transmit power p_n	30 dBm
Bandwidth between users and vehicles	20 MHz
Noise power N_0	−114 dBm
Sharable resource capability θ_k	370–770 MB

3.2. Task Offloading Problem Based on Service-Aware Matching

The primary objective of this problem is to minimize the overall delay and increase the number of successful matches between vehicular fog nodes and users for task offloading. This can be achieved using a service-aware matching algorithm, where each user constructs a preference list of vehicular fog nodes based on multiple factors, including their delay tolerance, communication radius, total delay (including transmission and processing delays), residence time of vehicles within the communication range, and the amount of resources shared by the vehicular fog node. According to the users' Quality of Service (QoS) requirements, task offloading decisions and the allocation of fog node resources are then determined.

4. PERFORMANCE EVALUATION

We consider fifteen different topologies. In one scenario, 30 users and 20 vehicular fog nodes are randomly distributed within a cell of radius 1000 m. In another scenario, 20 users and 20 vehicular fog nodes are deployed in the same manner. For each topology, the contract optimization problem is first solved to establish optimal contracts between the base station and the vehicular fog nodes. After solving the contract problem, the amount of resources shared by vehicular fog nodes, the rewards received, and the profits of both the base station and vehicular fog nodes are obtained for both complete-information and information-asymmetric scenarios.

In the second stage, the task offloading problem between users and vehicular fog nodes is solved using the service-aware adaptive matching method as described in Algorithm 1.

4.1. Contract Solution and Incentive Mechanism

In this section, the contracts under the information-asymmetric scenario are compared with the ideal complete-information scenario, in which the base station has full knowledge of the types of vehicles present in the network. The comparison is performed in terms of shared resources, rewards, and profits of both the base station and vehicular fog nodes.

Figures 1 and 2 illustrate the amount of computational resources shared and the corresponding rewards for different vehicle types in both complete-information and information-asymmetric scenarios. As observed, the shared computational resources and rewards increase uniformly with vehicle type. Moreover, the results show that under the complete-information scenario, the amount of shared resources is higher, as the base station has precise knowledge of the available resources of vehicular fog nodes. Since the available resources equal the shared resources in this scenario, the reward assigned to each vehicle is exactly 1 regardless of the vehicle type. This allows the base station to maximize its profit by increasing the shared resources.

Figures 3 and 4 depict the profits of the base station and vehicular fog nodes versus vehicle type. Numerical results indicate that information asymmetry benefits the vehicular fog nodes, as it protects them from over-utilization by the base station and guarantees their profit. In contrast, under the complete-information scenario, the base station can design contracts such that the profit of each vehicle is zero, thereby maximizing its own profit relative to the asymmetric scenario. The total profit of the base station and vehicular fog nodes is illustrated in Figure 5.

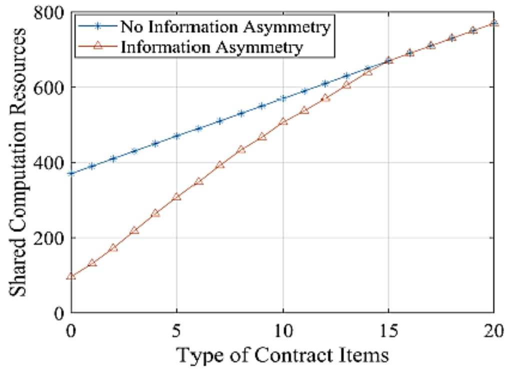


Fig. 1. Number of computational resources shared by vehicular fog nodes.

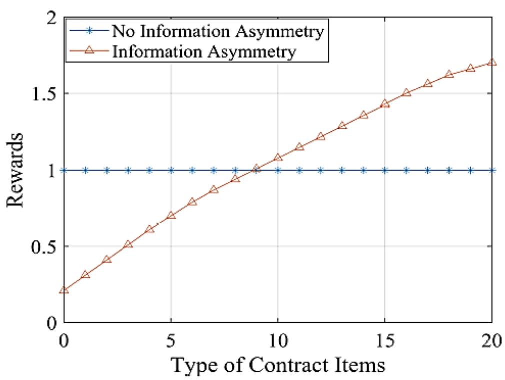


Fig. 2. Rewards received by vehicular fog nodes.

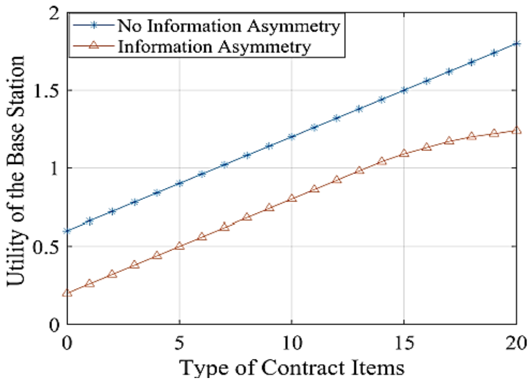


Fig. 3. Profit of the base station for different types of vehicles.

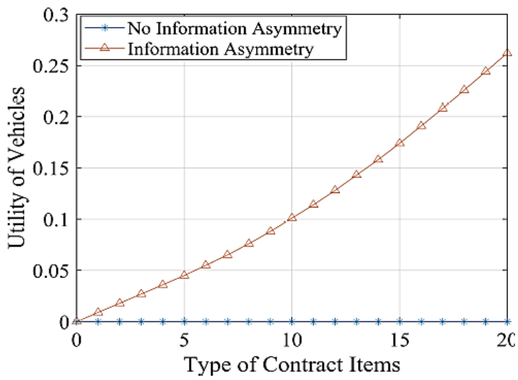


Fig. 4. Profit of vehicular fog nodes for different vehicle types.

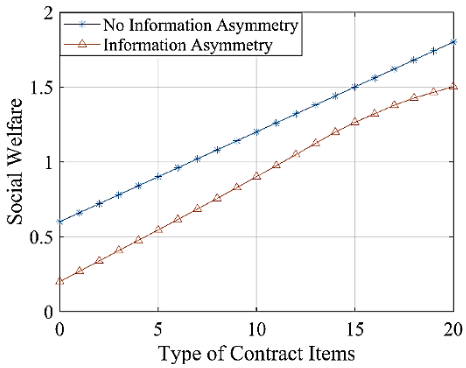


Fig. 5. Total profit of the base station and vehicular fog nodes across all vehicle types.

4.2. Task Allocation Based on Service-Aware Matching and Comparison with Benchmark Scheme

To evaluate the effectiveness of the proposed service-aware stable matching scheme for task allocation between users and vehicular fog nodes, we compare it with a benchmark method based on price-increasing stable matching [21]. In the benchmark scheme, a two-sided matching is performed between users and vehicular fog nodes for task offloading and resource allocation. Initially, each user is temporarily matched with available vehicular fog nodes, and the transmission and processing delays are calculated. Then, users construct a priority list of fog nodes based on these delays and submit requests to higher-priority nodes. A price-based stable matching algorithm is subsequently applied to resolve conflicts among users, where a server incrementally increases the offered matching price for competing users. Users update their priority lists according to the new costs and may propose to other servers. This iterative process continues until a stable matching between users and vehicular fog nodes is established for task offloading.

In the proposed service-aware matching algorithm, users are first prioritized into four groups based on their requested Quality of Service (QoS) and delay tolerance. Each user is temporarily matched with the available vehicular fog nodes, and the transmission and processing delays are calculated. Subsequently, users construct priority lists of fog nodes based on these delays and submit requests to the highest-priority nodes.

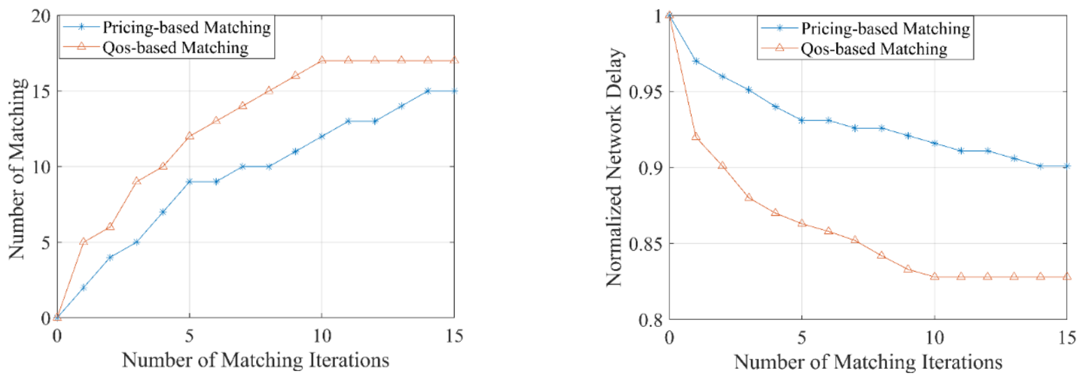
Unlike the benchmark method, our algorithm uses QoS criteria to resolve conflicts when multiple users request the same fog node. The node serves the user with a higher QoS priority, and the fog node is removed from the lower-priority users' lists. This approach ensures that the network delay for high-QoS users is minimized and reduces the overall network delay. When conflicts occur between users with identical QoS priorities, the node matches with the user whose computational task demands more resources, optimizing the use of shared fog resources and maximizing base station profitability. The process is repeated until all tasks are successfully offloaded to fog nodes.

In simulations, vehicle speeds are randomly selected in the range [2 – 20 m/s]. Network delay is normalized by dividing by the maximum delay experienced when all tasks are processed at the base station, scaling the delay to the range [0 – 1].

Figures 6 and 7 show the number of successful matches and normalized network delay for the proposed algorithm compared to the benchmark across two network topologies with 30 and 20 users and 20 vehicular fog nodes. Numerical results indicate that the proposed method achieves stable matching within fewer iterations, with faster convergence. Moreover, the number of matches in our algorithm is approximately 10% higher than the price-based method, while the total network delay for the 20-user topology is reduced by about 8%.

Figure 8 compares the base station workload between the proposed method and the benchmark. For 20 users, the workload is reduced by approximately 6%, and for 30 users, it decreases by about 11%. This improvement results from the higher number of matches and faster convergence in the proposed algorithm, which collectively reduce network delay and offloaded task volume at the base station.

Figure 9 shows that both the number of iterations required for convergence and the network delay increase with the number of users. This is because competition among users intensifies as their number grows, making it essential to consider QoS requirements when resolving conflicts. Additionally, if the number of users significantly exceeds the number of vehicular fog nodes, more tasks must be processed by the base station, substantially increasing network delay and base station workload.



a b

Fig. 6. Network with equal numbers of vehicular fog nodes and users:
(a) Number of matches in the vehicular fog network, (b) Total network delay in the vehicular fog network.

Furthermore, the base station's profitability can be accurately measured after executing the proposed task allocation algorithm in this study and compared with the theoretical scenario. In the theoretical case, it was assumed that all users are matched with all available fog nodes in the environment, which is an unrealistic assumption. After executing the algorithm, it was observed that some fog nodes remained unused under the given conditions, and no matching was established between these nodes and users. This demonstrates that the algorithm realistically reflects resource utilization and task assignment in a dynamic vehicular fog environment.

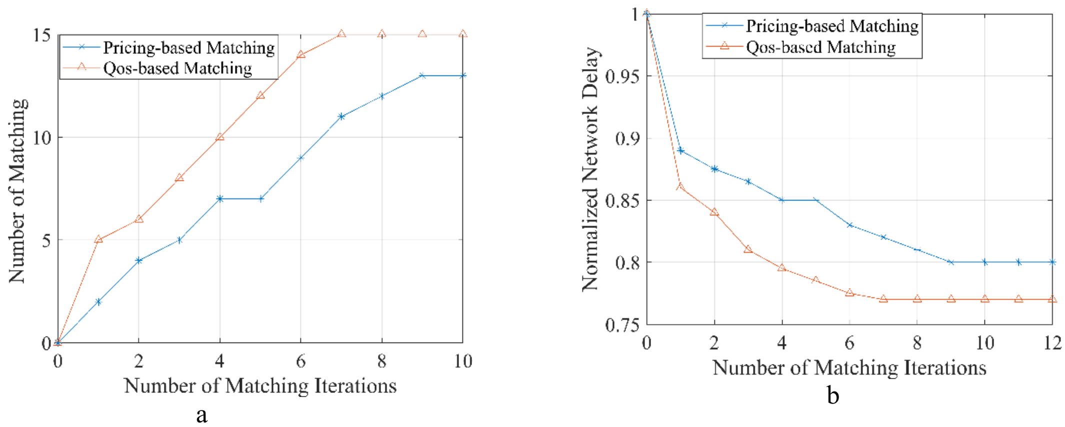


Fig. 7. Network with a higher number of users:
(a) Number of matches in the vehicular fog network, (b) Total network delay in the vehicular fog network.

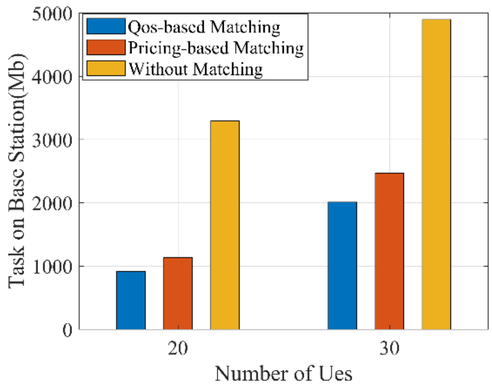


Fig. 8. Computational load on the base station.

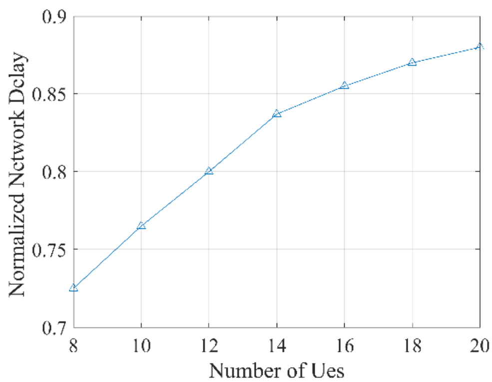


Fig. 9. Network delay versus the number of users.

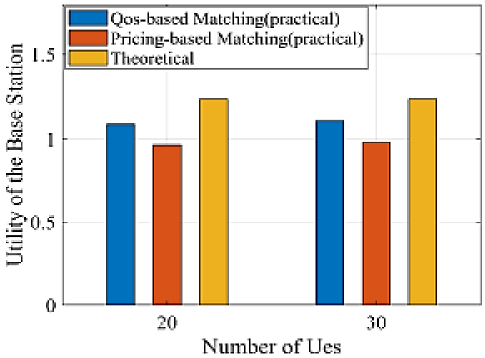


Fig. 10. Base station profitability after executing the proposed task allocation algorithm.

The results in Figures 7–10 demonstrate the effectiveness of the proposed QoS-aware task allocation algorithm. As the number of users increases, competition for fog nodes intensifies, leading to a gradual increase in network delay (Figure 9) and higher computational load on the base station (Figure 8). However, the proposed algorithm achieves a higher number of matches (Figures 7a and 7b) and faster convergence compared to benchmark approaches. Furthermore, Figure 10 shows that the base station's profitability after executing the proposed algorithm is realistically lower than the theoretical maximum, reflecting that some fog nodes remain unused due to dynamic network conditions, which is consistent with practical vehicular fog scenarios.

4.3. Task Allocation in Vehicular Fog Networks under Realistic Conditions

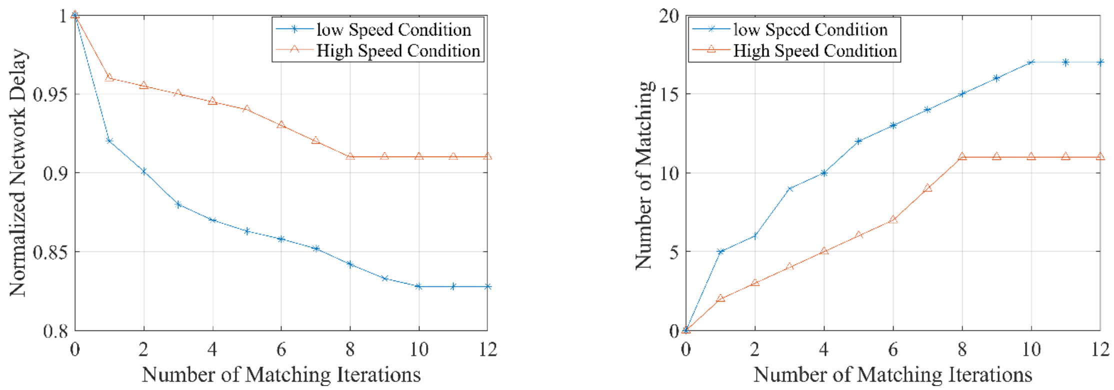
In the previous section, the task allocation problem was solved in a small-scale and constrained scenario, where the speed of vehicular fog nodes was limited and only a specific type of tasks was considered tasks that did not require continuous interaction between users and fog nodes during computation, with results returned to users after processing.

In this section, we consider a more realistic scenario by expanding the problem scale. The vehicular fog nodes move at higher speeds, randomly distributed between [10–50 m/s], and multiple types of tasks are considered, including those requiring continuous interaction with fog nodes during processing. The results are then compared with the previous scenario.

Figure 11 illustrates the number of matches and total network delay versus the number of iterations when the speed of vehicular nodes increases. As the speed increases, the residence time of fog nodes within the communication range of users decreases. Consequently, more user tasks must be processed by the base station, the number of matches between users and fog nodes decreases, and convergence slows down. Overall, the total network delay increases by approximately 9%.

Figure 12 shows the results for the second type of tasks those requiring continuous interaction with fog nodes during processing. Compared to the first type of tasks (which only required transmission to fog nodes without continuous interaction), the total network delay increases by about 6%, and convergence is slower.

From this analysis, it can be concluded that the proposed QoS-aware task allocation method effectively increases the number of matches between users and vehicular fog nodes and accelerates convergence. At the same time, it helps reduce the total network delay and the computational load on the base station under more realistic and dynamic conditions.



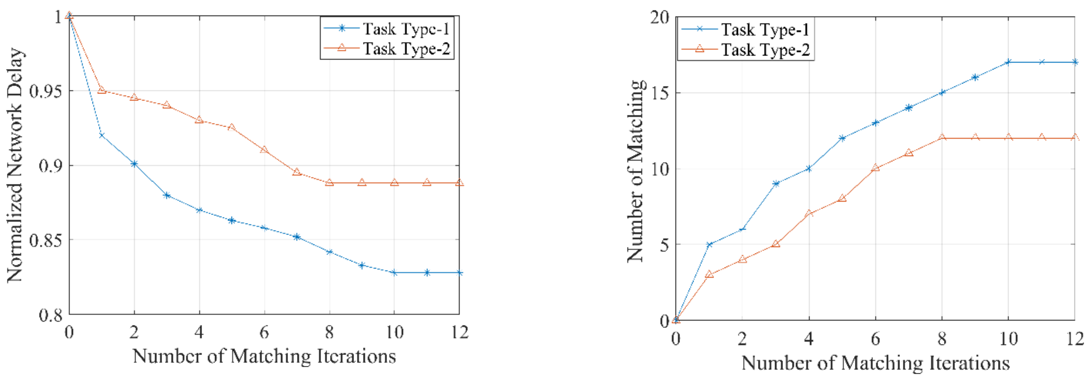
a b

Fig. 11. Effect of increased speed of vehicular fog nodes: (a) Number of matches in the vehicular fog network, (b) Total network delay.

Table 2. presents the total network delay for different numbers of users under each algorithm.

Number of Users	Total Delay (s)	Delay in Price-Based Algorithm (s)	Delay in Proposed Algorithm (s)
20	35.02	28.016	26.96
30	50.1	45.14	41.48

The results indicate that the proposed algorithm not only reduces the total network delay but also enhances the efficiency of task allocation in dynamic vehicular fog network scenarios.



a b

Fig. 12. Considering Type 1 and Type 2 tasks in the vehicular fog network: (a) Number of matches in the vehicular fog network, (b) Total network delay.

Table 3 illustrates the performance improvement of the proposed QoS-aware task allocation algorithm compared to the existing price-based algorithm in [12]. The improvements are reported in terms of number of matches, total network delay, and base station load.

Table 2. Performance Improvement of the Proposed QoS-Aware Algorithm

Number of Users	Improvement in Matches	Improvement in Delay	Improvement in Base Station Load
20	10%	8%	6%
30	10%	4%	11%

The results demonstrate that the proposed algorithm consistently improves task matching efficiency, reduces network delay, and decreases the computational load on the base station compared to the benchmark algorithm.

5. CONCLUSION

Fog computing serves as a bridge between cloud servers and edge devices, enabling processing, storage, decision-making, and data management not only in the cloud but also in locations closer to end devices. The integration of vehicular ad hoc networks (VANETs) with fog computing leads to vehicular fog networks, which provide reliable services for delay-sensitive applications. However, one of the main challenges in such networks is the limited fog computing capacity, especially under peak demand, which can severely degrade system performance. Therefore, utilizing the surplus computational resources of nearby vehicles is proposed as a cost-effective and demand-adaptive solution.

By aggregating computational resources from a large number of vehicles, network congestion during peak periods can be alleviated without deploying additional servers, enabling real-time computational scaling. This new computing paradigm vehicular fog computing acts as a complementary layer to edge and cloud computing in vehicular networks. The focus of this study has been on addressing the key challenges inherent in vehicular fog networks.

Following [12], the problem of computational resource allocation and task assignment was investigated through a contract-theoretic perspective. A contract-based incentive mechanism was proposed to motivate vehicles to share their resources. Furthermore, a QoS-aware task allocation algorithm was developed to efficiently assign user tasks to vehicular fog nodes. Simulation results demonstrated that the proposed approach reduces overall network delay, decreases base station overhead, lowers network complexity, and increases the number of successful task matches between users and vehicular fog nodes.

Declaration

We acknowledge that we used ChatGPT to enhance the academic writing of our manuscript while ensuring the originality and integrity of our work.

Transparency Statement

The data supporting this study are available upon reasonable request to the corresponding author, subject to ethical and confidentiality considerations.

Acknowledgments

We would like to express our gratitude to all individuals who contributed to this project.

Declaration of Interest

The authors declare that they have no competing interests.

Funding

This research received no specific grant from any funding agency, commercial, or not-for-profit sectors.

REFERENCES

[1] Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, Y., Zhou, P., Chai, Y., Caine, B., Vasudevan, V., Han, W., Xiao, J., Hu, Y., & Anguelov, D. (2020). Scalability in perception for autonomous driving: Waymo Open Dataset. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 2443–2451). IEEE. https://doi.org/10.1109/CVPR42600.2020.00252

[2] Mell, P., & Grance, T. (2011). *The NIST definition of cloud computing* (NIST Special Publication 800-145). National Institute of Standards and Technology. https://doi.org/10.6028/NIST.SP.800-145

[3] Adnan, U. K., & Ahmed, N. (2022). Thermal enhancement and entropy investigation in dissipative ZnO-SAE50 under thermal radiation: A computational paradigm. *Waves in Random and Complex Media, 35*, 3573–3588. https://doi.org/10.1080/17455030.2022.2053243

[4] Bennis, M., Debbah, M., & Poor, H. V. (2018). Ultrareliable and low-latency wireless communication: Tail, risk, and scale. *Proceedings of the IEEE, 106*(10), 1834–1853. https://doi.org/10.1109/JPROC.2018.2867029

[5] Bilò, V., Moscardelli, L., & Vinci, C. (2024). Uniform mixed equilibria in network congestion games with link failures. *Mathematics of Operations Research, 49*, 509–535. https://doi.org/10.1287/moor.2023.1365

[6] Qi, Z., Yinfa, Z., Balin, T., Ding, D., & Shuai, R. (2020). Research on the calculation method of transmission bandwidth demand of private network communication hub station. In *2020 International Conference on Wireless Communications and Smart Grid (ICWCSG)* (pp. 32–36). IEEE. https://doi.org/10.1109/ICWCSG50807.2020.00016

[7] Wu, D., Zhang, P., He, Y., & Luo, X. (2023). A double-space and double-norm ensembled latent factor model for highly accurate web service QoS prediction. *IEEE Transactions on Services Computing, 16*, 802–814. https://doi.org/10.1109/TSC.2022.3178543

[8] Sharma, T., Mangla, T., Gupta, A., Jiang, J., & Feamster, N. (2023). Estimating WebRTC video QoE metrics without using application headers. In *Proceedings of the 2023 ACM Internet Measurement Conference*. Association for Computing Machinery. https://doi.org/10.1145/3618257.3624828

[9] Sciangula, G., Casini, D., Biondi, A., Scordino, C., & Natale, M. (2023). Bounding the data-delivery latency of DDS messages in real-time applications. In *2023 IEEE Real-Time Systems Symposium* (pp. 9:1–9:26).

[10] Mach, P., & Becvar, Z. (2017). Mobile edge computing: A survey on architecture and computation offloading. *IEEE Communications Surveys & Tutorials, 19*(3), 1628–1656. https://doi.org/10.1109/COMST.2017.2682318

[11] Kashani, M. H., & Mahdipour, E. (2023). Load balancing algorithms in fog computing. *IEEE Transactions on Services Computing, 16*, 1505–1521. https://doi.org/10.1109/TSC.2022.3174475

[12] Yousefpour, A., Fung, C., Nguyen, T., Kadiyala, K., Jalali, F., Niakanlahiji, A., Kong, J., & Jue, J. P. (2019). All one needs to know about fog computing and related edge computing paradigms: A complete survey. *Journal of Systems Architecture, 98*, 289–330. https://doi.org/10.1016/j.sysarc.2019.02.009

[13] Kai, K., Cong, W., & Tao, L. (2017). Fog computing for vehicular ad hoc networks: Paradigms, scenarios, and issues. *Journal of China Universities of Posts and Telecommunications, 23*, 56–65. https://doi.org/10.1016/S1005-8885(16)60021-3

[14] Zhou, Z., Liu, P., Feng, J., Zhang, Y., Mumtaz, S., & Rodriguez, J. (2019). Computation resource allocation and task assignment optimization in vehicular fog computing: A contract-matching approach. *IEEE Transactions on Vehicular Technology, 68*(4), 3113–3125. https://doi.org/10.1109/TVT.2019.2894851

[15] Zhou, Z., Liao, H., Wang, X., Mumtaz, S., & Rodriguez, J. (2020). When vehicular fog computing meets autonomous driving: Computational resource management and task offloading. *IEEE Network, 34*, 70–76. https://doi.org/10.1109/MNET.001.1900527

[16] Ning, Z., Huang, J., & Wang, X. (2019). Vehicular fog computing: Enabling real-time traffic management for smart cities. *IEEE Wireless Communications, 26*, 87–93. https://doi.org/10.1109/MWC.2019.1700441

[17] Ning, Z., Dong, P., Wang, X., Guo, L., Rodrigues, J. J. P. C., Kong, X., & Huang, J. (2019). Deep reinforcement learning for intelligent Internet of Vehicles: An energy-efficient computational offloading scheme. *IEEE Transactions on Cognitive Communications and Networking, 5*, 1060–1072. https://doi.org/10.1109/TCCN.2019.2930521

[18] Zhou, S., Sun, Y., Jiang, Z., & Niu, Z. (2019). Exploiting moving intelligence: Delay-optimized computation offloading in vehicular fog networks. *IEEE Communications Magazine, 57*, 49–55. https://doi.org/10.1109/MCOM.2019.1800230

[19] Ouyang, T., Zhou, Z., & Chen, X. (2018). Follow me at the edge: Mobility-aware dynamic service placement for mobile edge computing. *IEEE Journal on Selected Areas in Communications, 36*(10), 2333–2345. https://doi.org/10.1109/JSAC.2018.2869954

[20] Alameddine, H., Assem, H., Sharafeddine, S., Sebbah, S., & Ayoubi, S. (2019). Dynamic task offloading and scheduling for low-latency IoT services in multi-access edge computing. *IEEE Journal on Selected Areas in Communications, 37*(3), 668–682. https://doi.org/10.1109/JSAC.2019.2894306

[21] Huang, C., Lu, R., & Choo, K. K. R. (2019). Vehicular fog computing: Architecture, use case, and security and forensic challenges. *IEEE Communications Magazine, 57*, 105–111. https://doi.org/10.1109/MCOM.2017.1700322