

Parked Vehicles-Assisted Fog Computing: Review of Task Offloading Approaches and Strategies

Alla Abbas Khadir

*College of Computer Science and Information
Technology, University of karbala,
karbala, Iraq*

alla.a@uokerbala.edu.iq

<https://Orcid.org/0000-0003-1781-7810>

Baydaa Fadhil Dhahir

*College of Computer Science and Information
Technology, University of karbala,
karbala, Iraq*

baydaa.f@uokerbala.edu.iq

<https://Orcid.org/0000-0003-1781-7810>

DOI: <http://dx.doi.org/10.31642/JoKMC/2018/130203>

Received Dec. 5, 2025. Accepted for publication Jun. 11, 2026

Abstract—Recently, parked vehicles (PVs) have emerged as a new computing paradigm to assist the limited resources of the fog node (FN) in the task offloading process. The underutilized resources of PV have become a primary feature that attracts many researchers to propose various task offloading schemes and approaches. Existing work primarily focuses on exploiting PV resources to optimize delay, energy, and cost in the task offloading process. This review article describes related work published between 2018 and 2025. Specifically, we have recognized and classified the diverse approaches and task offloading schemes that have been established according to their objectives. In addition, we discuss some research challenges and open issues in collaborative FN-PV task offloading schemes.

Keywords—Fog computing, Parked vehicles, Task offloading, Delay, Cost, Energy, Utilization.

I. INTRODUCTION

The global number of the Internet of Things (IoT) is increasing dramatically and is estimated to be around 37.1 billion devices by 2032[1]. Simultaneously, the ever-increasing generation of various tasks of IoT requires different computing resources in the future. Therefore, fog computing is considered a suitable computing offload destination to process tasks offloaded from the IoT [2]. However, the limited computing resources of the FN lead to re-offloading these tasks to other computing offload destinations such as nearby FN or the cloud [3]. This offloading approach introduced a new problem for the FN, namely additional latency, energy, and cost [4]. To address this problem, the researchers investigated an alternative supported computing paradigm for the FN, specifically PVs [5]. They observed that the PV always has idle and rich embedded computing resources to process offloaded tasks. Recently, works have studied the collaborations either between FN and PV directly [6] or by a third party such as SDN [7] to optimize (i.e., maximize or minimize) the objective function. According to the objective function, different efficient offloading approaches and strategies are designed for the FN-PV environment. However, the proposed offloading approaches face some challenges, such as the residual energy and remaining parking time of the PV.

These challenges are not considered in most works [6], [8], [9], [10], whereas they are included in some works [11], [12], [13]. In this review, we focus on the recent existing works that study the extension capacity of the FN by the PV, and explore the offloading approaches proposed in the literature. The main contributions of this review can be summarized as follows:

- We provide a complete overview of the FN-PV offloading schemes by introducing the main components that comprise the scheme.
- We introduce the concept of task offloading and explore the diverse common types of FN task offloading based on the offload destination.
- We present a classification for the approaches that have recently been proposed for the FN-PV offloading environment based on the desired objectives.
- Finally, we explore the open challenges and future research directions in the FN-PV offloading scheme.

This paper is organized as follows. In Section 2, we will present an overview of the main components of the recently proposed FN-PV offloading scheme. In Section 3, we will

classify the approaches that are proposed for the FN-PV offloading scheme according to the desired objective. In Section 4, we explore future research directions in the FN-PV offloading scheme and identify the open challenges. Finally, we conclude our paper in Section 5.

A. Background on The Main Components of FN-PV Offloading Schemes

There are several FN-PV offloading schemes that are proposed, but most of them include typical hardware components such as the Internet of Things (IoT), cloud computing, fog computing, and parked vehicles. This will be discussed in more detail below.

- **IoT:** Internet of Things (IoT) is a network of physical appliances such as vehicles, cell phones, cameras, and actuators. These objects are embedded with sensors, software, and other technologies that allow them to send and receive data with other devices over the internet. Recently, the number of IoT devices has increased dramatically. Furthermore, the IoT generates large amounts of traffic quickly that require additional processing and storage resources [14] that are not available on IoT devices. Therefore, limited computing resources are considered as an essential problem in the IoT [2]. In general, to overcome this problem, IoT offloads some or all tasks to more resource-intensive computing, such as fog computing, to help with processing [15], [16]. Many proposed works have focused on the use of FN resources to process offloaded IoT tasks that are presented in different mechanisms [17],[18],[19], [20].
- **Cloud computing :** Cloud computing is presented as a service provisioning model that provides ubiquitous on-demand access to a shared pool of physical and virtualized computing, communication, and storage resources [21]. It allows high utilization of resources with large scalability and flexibility. Cloud computing offers three diverse service provisioning models: infrastructure as a service (IaaS), platform as a service (Paas), and software as a service (SaaS) [22], [23]. Cloud resources are centralized and located remotely from IoT, which leads to challenges such as limited bandwidth, high latency, and an inability to effectively meet the low-latency requirement [24]. Accordingly, cloud computing cannot support the huge amount of data generated by IoT, which needs to be transferred and processed in real time. Therefore, another promising technology is needed, in which computing services are located close to the IoT in the edge network for processing tasks and storing data, then transmitting the IoT data to the cloud for significant analysis [25]. Fog computing was introduced as a promising and novel paradigm for extending cloud services [26].
- **Fog computing :** Fog computing has emerged as a complement to the cloud by providing a platform of computing, storage, and communication services at the edge of the network [27]. This proximity of the FN to

the end devices, such as IoT, vehicles, etc., can reduce the communication delay of real-time applications by keeping the data and computation close to the end devices [28]. Fog computing consists of heterogeneous resources of edge network devices such as APs, switches, routers, gateways, BSs, etc., which are named FN. FN has limited computing resources [29], which will be a bottleneck for task computing in high-density scenarios [30]. Overloaded FN sends tasks to other low-loaded FNs to meet the required deadline offloaded tasks can be processed or offloaded many times for nearby FNs or cloud [31]. Therefore, FN required an efficient management approach to increase its utilization and decrease the response time of the offloaded tasks [32]. Many research works have been proposed to increase the efficiency and make fog computing more powerful [7],[33],[34],[35],[36],[37].

- **Parked vehicles :** In our daily life, there are many vehicles parked on streets and parking lots that are stopped for over 95% of the day [11]. These vehicles have abundant and idle computing and storage resources, which can be used to support the FN resources [38]. The PVs can join the FN system and share their idle computing resources to process offloaded tasks that are assigned by the FN, as well as they can forward tasks as the FN [5]. Therefore, many existing works [39], [40],[10],[41] are motivated to utilize PV to support FN resources in computation offloading. In this review, we study the coordination between PVs and FN in terms of provisioning resources in computation offloading, where this is the first study to review related works on task offloading in FN-PV collaboration.
- **Task offloading:** The task offloading method is to transfer the execution of a task from a resource-constrained node to a remote or nearby resourceful node [4]. Task offloading is considered a main strategy in FN to address its limited resources [42]. Depending on the offload destination, the offloading can be classified into vertical and horizontal offloading [3]. Vertical offloading is transferring of the processing task from the FN to the cloud or UAV, while horizontal offloading involves transferring the execution task from the FN to the nearby FN, MV, or PV. Figures 1 and 2 illustrate the IoT-FN- PVs architecture and options of task offloading for the FN, respectively.

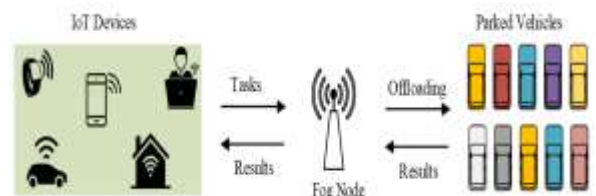


Fig. 1: IoT-FN-PVs architecture.

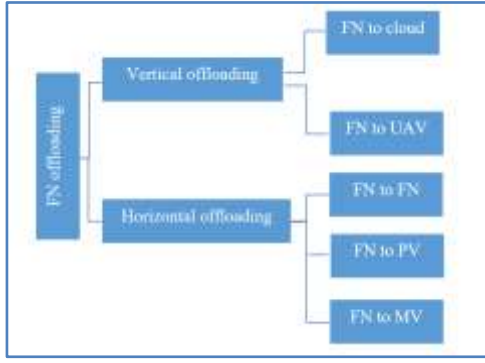


Fig. 2: FN offloading approaches.

Both types of task offloading can be full or partial [43]. In full task offloading, the FN sends the whole task to another offload destination. On the other hand, in partial task offloading, the FN sends only a portion of the task to another offload destination. Generally, tasks that are offloaded can significantly save energy, reduce delay, and earn cost [26]. However, horizontal offloading has better performance than vertical offloading [44], [45]. The reason is that the performance of cloud services is limited by several challenges, such as the remote distance of cloud servers and network congestion [46]. Due to its importance and impact, leveraging FN and PV has attracted considerable interest from numerous scholars. Here, we will present and discuss the related FN-PV collaborative computation offloading schemes. In addition, we study the proposed offloading approaches with different objectives. Table I indicates the abbreviations applied in this article.

TABLE 1. EXISTING WORK ON FN-PV TASK OFFLOADING APPROACHES

Abbreviations	Description
AP	Access Point
BS	Base Station
FN	Fog Node
ICV	Intelligent connected vehicle
IoT	Internet of Things
IoV	Internet of Vehicle
MV	Mobile Vehicle
MD	Mobile Device
OLD	OffLoad Destination
PV	Parked Vehicle
RL	Reinforcement Learning
SDN	Software Defined Network
SP	Service Provider
UE	User Equipment
UAV	Unmanned Aerial Vehicle
MEC	Mobile Edge Computing
VEC	Vehicular Edge Computing
RSU	Roadside Unite
OLD	OffLoad Destination

ECP	Edge Crowdsourcing Platform
DRL	Deep Reinforcement Learning
DQN	Deep Q-Network
QoS	Quality of Service
V2I	Vehicle-to-Infrastructure
V2V	Vehicle to Vehicle
PVEC	Parked vehicle edge computing
RSPV	RoadSide parked vehicles
CVEC	Collaborative Vehicular Edge Computing
MPVEC	Multiple parked vehicle-assisted edge computing
IaaS	Infrastructure as a service
PaaS	Platform as a service
SaaS	Software as a service
LSAC	Lyapunov-based Soft Actor-Critic
BLP	Binary-Linear Programming
MILP	Mixed Integer Linear Programming
MINLP	Mixed Integer Non-Linear Programming
HHDRl	hybrid hierarchical deep reinforcement learning
TOSRA	task offloading, scheduling, and resource allocation

II. TAXONOMY OF OFFLOADING APPROACHES FOR FN-PV SCHEMES BASED ON THE DESIRED OBJECTIVE

Now, we describe numerous task offloading approaches for FN-PV schemes that have been recently proposed in the literature. Based on the optimization objectives in the related works, we classify the existing research into five categories: Table II summarizes the reviewed studies.

A. Minimizing Offloading Delay

Minimizing delay in the task offloading process is an optimization challenge that has attracted significant attention from several scholars. Reference [9] formulated the offloading problem to provide real-time responses for vehicular applications, where the service delay network is minimized by utilizing the computing resources of PV. The offloading problem has been solved by a heuristic algorithm combined with RL. The results showed that the proposed algorithm can easily be adjusted to make resource allocation decisions accordingly and achieve higher service satisfaction compared to traditional resource allocation algorithms. The heterogeneity and undefined parking duration of PVs were motivated to make optimal service migration decisions in PVEC networks [47]. The authors formulated the service migration process as a MINP optimization problem to minimize the average service latency in PVEC. They proposed a two-stage service migration algorithm to solve the formulated problem, which divides the optimization problem into the service migration between SP and the selected PVs. By utilizing the expected parking duration of PVs, the service migration between SPs is transformed into an online problem based on Lyapunov optimization. A modified Hungarian

algorithm is proposed to select PVs for migration. Finally, the effectiveness of the proposed algorithm is verified by simulation based on the real-world vehicle traces. Reference [7] proposed an SDN-based offloading selection policy to choose an OLD, where the OLD can be PV, MV, or cloud. It employs the BLP technique to determine the optimal OLD for processing the offloaded task that can satisfy the required deadline. The communication time and the computation time of the offloaded task are minimized. The simulation results demonstrated that the proposed policy improves the performance of the FN by providing less response time and significantly outperforming baseline policies. Reference [41] suggested a collaborative computation framework and a metaheuristic algorithm to optimize the acceptance ratio and the execution delay for PV. This framework uses the Kubernetes-based container orchestration, which allows PV to handle several task replicas simultaneously and efficiently. Additionally, it ensures continuity of service by relaying tasks between the active nodes. The simulation results showed that the proposed computing framework remarkably increases acceptance of offloaded tasks and improves the average task offload delay. Reference [48] proposed a joint task offloading and resource allocation scheme for a PV and MV-enabled MEC scenario, in order to minimize the total priority-weighted task processing delay for all devices. A generalized Benders decomposition with reformulation-linearization and a two-stage heuristic algorithm are developed to solve the problem with low computational complexity. Simulations of seven scenarios confirm the superiority of the proposed scheme over three existing schemes. Reference [49] used PV as a supporting FN to decrease the number of offloaded tasks and increase the capacity of FN in the IoV network. In addition, the author proposed a novel load-balancing method on the basis of a proactive approach, which consequently helped in reducing the task latency. The proposed method can balance the local and global load using local fog managers and the SDN controller, respectively. Simulation results validated the effectiveness of the proposed method in terms of average response time, achieving the deadline, bandwidth consumption, and resource utilization. Parking edge computing is a new concept introduced in [11] to expand the VEC service. These PVs are transformed into virtual edge servers within the VEC infrastructure for task execution. Furthermore, the authors proposed a task scheduling algorithm to select the edge server and resource allocation by considering the remaining deadline of the task and the battery energy of PV. Reference [50] focused on the variability in PV resources and the parking time problem, which leads to significant challenges in designing an efficient offloading mechanism. To address this, the authors formulate an NP-hard optimization problem to minimize delay while managing service interruption constraints. They proposed an LSAC algorithm to solve it. Experiments show that the proposed algorithm improves offloading performance by reducing both interruption rate and delay. Reference [51] proposed a PV-assisted VEC architecture where a multi-agent collaboration has been considered, which includes intelligent vehicles and edge servers. Also, a Stackelberg game model was constructed based on the interaction between the requesting vehicles and

an SP. This model is solved by designing a distributed gradient-based equilibrium algorithm where the final decision is obtained through mutual communication. Simulation results confirmed that the proposed algorithm outperforms existing algorithms in improving the utility of the service provider. An offloading framework of dependent task scheduling is designed in [52], in which the PVs are clustered and cooperate with the RSU. Also, a DRL algorithm is proposed to minimize the average completion time of the application. The proposed algorithm is evaluated, and it can achieve better task processing time and task execution success rates compared to conventional algorithms.

B. Minimizing Energy Consumption

Minimizing energy consumption during the task offloading process is a main challenge that has been addressed in a few studies. In [53], both PVs and MVs are leveraged as FNs, where a three-layer network model is constructed in IoV. The optimization problem is formulated to reduce energy consumption during task offloading. To reduce the computational complexity of the formulated problem, it is divided into two parts which are flow redirection and offloading decisions. A DRL-based scheme is designed to solve the optimization problem. In [54], a computational offloading strategy based on a “deep Q-learning algorithm” was proposed to reduce the offloading energy consumption in VEC networks. Two offloading approaches have been considered: which are RSU-offloading tasks via V2I communication and service vehicle-offloading tasks via V2V communication. Although the RSU and the service vehicle occupy different positions, the author modelled both as a unified service node. Simulation results proved the effectiveness of the proposed strategy in terms of reducing total offloading energy consumption.

C. Minimizing Offloading Cost

Reducing the task offloading cost in terms of data transmission cost, task computation (processing) cost, or both has attracted many authors. To minimize the total cost for users in computation offloading, Reference [39] proposed a PVEC architecture, where the idle resources of PVs are fully utilized and cooperate with the edge servers to run mobile applications. In addition, a resource scheduling algorithm based on the Stackelberg game is presented to determine the allocation of the workload among PVs with the objective of reducing the total cost of users. Numerical results indicated that the Stackelberg game approach is effective and efficient in PVEC. Also, the previous approach (i.e., Stackelberg game) is used in [55] to solve the task assignment problem in the PVEC scheme. Accordingly, the SP will minimize the monetary cost, while the PVs maximize revenue. Finally, the numerical results indicate that the proposed scheme is effective and outperforms the existing work in terms of reducing the overall cost of SP. Reference [38] proposed a new offloading scheme where the RSPV is utilized as a new computing paradigm to process the offloaded task of the IoV. The assignment task problem is formulated as an MILP model to minimize the transmission and processing cost of the offloaded tasks under the required budget and deadline. The

simulation results validated the effectiveness of the proposed offloading scheme. To reward the owner of PV for utilizing computing resources in supporting task executions within MEC, a dynamic pricing strategy is proposed by [56]. The proposed strategy can minimize the average cost of the MEC system under QoS constraints by adjusting the price constantly based on the current system state. The simulation results demonstrated the effectiveness of the proposed strategy.

Reference [57] designed an incentive-hierarchical pricing mechanism to encourage PVs for sharing their idle resources in task offloading. Also, the interaction between the PVs and the task vehicle is modeled as a Stackelberg game and analyzed using backward induction. Furthermore, a genetic algorithm is developed to achieve the Stackelberg equilibrium and determine the optimal strategy for task vehicles. Extensive simulations demonstrate the effectiveness of the proposed offloading mechanism.

D. Minimizing Utilization

Increasing the resource utilization of the FN is another important objective, aiming to increase the number of computing services over the FN [16]. For improving the task execution of PVs with fast startup, minimal overhead, and secure isolation, containerization is applied in [58]. The authors introduced a task-container matching market that dynamically offloads services. By analyzing parking behavior and resource availability, the system selects appropriate PVs for reliable task processing. They employed a utility-based preference modelling and matching game approach to optimally assign tasks to PV-hosted containers. The numerical results showed significant improvements in task completion and acquired a higher utility. A new framework named CVEC is introduced in [59] that integrates the PVs with MEC. In CVEC, the system task allocation is optimized by designing an optimal contract-based incentive mechanism to schedule PVs on demand and to handle offloading tasks. Also, the optimal contract is used to maximize the utility of the user under information asymmetry. Numerical results are presented to validate the effectiveness and efficiency of the proposed scheme. A partial offloading strategy for a multi-user PVMEC scheme is studied in [60], where the tasks of MD are partially offloaded to the MEC or a close PV. The system utility maximization problem is formulated by considering the offloading decisions, offloading ratio, and resource allocation. The offloading decision-making problem is modelled using a game-theoretic approach to reduce complexity. Furthermore, the optimal offloading ratio and resource allocation are determined using a sub-gradient method. Simulation results demonstrate that the proposed approach can significantly improve the system utility over traditional schemes. The PVEC paradigm is exploited in [61] to propose a new partial offloading scheme. In this case, the resources of both MD and PV are used for parallel task execution in PVEC. Here, the offloading utility of MDs is maximized, where the utility represents the difference between the benefit of latency reduction, computing utilization cost, and networking resources. The simulation results indicated that the proposed offloading scheme can improve the total offloading utility compared to the baseline offloading schemes. Reference [62]

proposed a VEC framework based on SDN that uses PV to assist edge servers in executing offloaded tasks. An ECP is designed to manage the idle computation resources of PV. The optimal strategy for the edge server and ECP is selected through a genetic algorithm-based game-strategy solving algorithm. The proposed solution outperforms traditional solutions without ECP, producing higher edge server utility and faster convergence.

E. Minimizing Energy and Delay

Multi-objective optimization during the task offloading process in the FN-PV scheme has been conducted in a few studies. In this subsection, we explore the studies that proposed offloading approaches that minimize energy and delay. Reference [6] proposed an MPVEC offloading scheme to reduce the load on MEC devices. After that, the load balancing and task offloading problem is formulated to minimize the energy consumption under a delay constraint. In addition, they proposed a multiple-offloading node selection algorithm to select the appropriate parked vehicles when executing the offloaded tasks. Furthermore, a workload allocation strategy based on the idea of a dynamic game is used to optimize system performance and consider the load balancing at the same time. Numerical results indicate that the proposed offloading strategy in the MPVEC scheme effectively reduces the system energy under delay constraint while achieving system load balancing. In [63], the task offloading collaboration between the MEC server and PVs is studied to reduce the delay of offloaded tasks and energy consumption of mobile devices, which improved task offloading efficiency and reduced the burden on the MEC server. In other words, they proposed a resource management scheme and an execution model of computation and communication in PVMEC, which is reviewed. The optimization problem is formulated into an MINLP and decomposed into two sub-problems with lower complexity. The simulation results demonstrate the effectiveness of the proposed solution. Reference [64] explores VEC to improve UE performance by utilizing idle resources from PVs and RSUs. It introduces a joint offloading architecture and models computation offloading as a dynamic multi-objective optimization problem. To address environmental uncertainty, the study proposes a dynamic multi-objective evolutionary algorithm with multi-strategy fusion response, incorporating LSTM-based prediction and adaptive strategies. Experiments show the approach effectively balances latency and energy consumption while ensuring solution convergence and feasibility.

Reference [65] introduced a vehicle-road collaborative edge computing network for efficient task offloading in the ICV-MEC network. A TOSRA optimization problem is formulated to minimize the overall time and energy consumption costs of the dynamic system. To solve the complex optimization problem, a DRL-based HHDR algorithm is proposed. The stability and convergence of the proposed algorithm are proved and compared to current benchmark algorithms.

TABLE 2. EXISTING WORK ON FN-PV TASK OFFLOADING APPROACHES

Optimization objective	Ref.	Algorithm type	Offloading approach	Weaknesses
Delay	[7]	Mathematical optimization	SDN has a global view of the entire network and requests from the overloaded FN to offload tasks to a suitable PV.	Energy consumption of PVs did not considered
	[9]	Mathematical (heuristic)	A dynamic resource allocation approach combining heuristic methods and reinforcement learning based on parking status.	Energy consumption of PVs did not considered
	[11]	task scheduling	Organizing PVs into clusters, then transforming them into virtual edge servers within the VEC infrastructure.	Transmission time of the offloaded task did not considered.
	[41]	Metaheuristic	A collaborative computation framework uses the Kubernetes-based container orchestration and a metaheuristic algorithm.	NP-hard problem
	[47]	Lyapunov optimization	Making optimal service migration decisions in PVEC networks	NP-hard problem
	[48]	Two-Stage Heuristic	A joint task offloading and resource allocation scheme for a PV and MVs	Energy consumption of PVs is neglected
	[49]	Mathematical optimization	A proactive load-balancing approach that reduces the number of offloaded tasks and improves the capacity of FN.	More parameters has been neglected such as execution time, response time, deadline, cost of the offloaded task as well as the energy of PVs.
	[50]	Lyapunov-based Soft Actor-Critic (LSAC)	Considering the heterogeneity of the computing resources among various PVs and the dynamic nature of their parking states.	NP-hard problem
	[51]	Game theory	Stackelberg model is based on the interaction between requested vehicles and an SP that assigns the resources of VEC and PV.	High complexity of the formulated problem and deadline of the offloaded task is neglected.
Energy	[53]	Double Deep Q Network (DDQN)	A three-layer offloading framework to minimize the overall energy consumption while satisfying the delay constraint in an intelligent IoV network.	High complexity formulated problem and the remaining energy of EVs is neglected.
	[54]	Deep Q Network (DQN)	DQN-based partial offloading approach for VEC network to minimize the total energy consumption.	Partial offloading consumes high energy due to continuous data transmission and processing.
Cost	[38]	Mathematical optimization	A mathematical optimization model is used to assign offloaded tasks to the best RSPVs.	The deadline of the task and energy of PVs have been neglected
	[57]	Genetic algorithm	An incentive-hierarchical pricing mechanism to encourage PVs for sharing their idle resources in task offloading.	The transmission cost of the offloaded task did not consider in the system model.
	[39]	Game theory	A response scheduling algorithm based on a Stackelberg game is formulated to implement resource allocation among PVs.	A centralized architecture is proposed, which is exposed to point of failure problem.
	[55]	Game theory	SP represents a leader to offer incentives, whereas the PVs represent followers to participate in computation offloading concerning the specified rewards	High complexity problem
	[56]	Mathematical optimization	A stochastic process is used to model the arrival and departure rates of PVs based on historical information.	Optimizing the purchase of computing resources only, without optimizing energy pricing for the user.
Utilization	[58]	Matching game	A task-container matching market provides on-demand offloading services.	The impact of service offloading and QoS has been ignored.
	[59]	Contract theory	An optimal contract-based incentive mechanism to schedule PVs on demand and to handle offloaded tasks.	The effect of the limited energy of PVs and residual resources of MECs on the system performance is neglected.
	[60]	Game theory	A partial offloading approach for multi-user PVMEC, where the tasks of MD can be partially offloaded to the MEC or a close PV.	Partial offloading suffers from complex decision-making and increases the risk of overhead and delay.
	[61]	Mathematical optimization	A partial offloading approach that allows MD to utilize local and remote resources for parallel task processing in PVEC.	Partial offloading suffers from complex decision-making and increases the risk of overhead and delay.
	[62]	Genetic algorithm	An ECP manages and integrates the idle computation resources of electric PVs to provide computation services for requesting vehicles.	The remaining energy of EVs did not considered in modeling.

Energy, Delay	[6]	Mathematical optimization	According to workload balance among computing nodes, a multiple offloading node selection approach selects appropriate PVs.	NP-hard problem, and scalability can be affected by the participant PVs
	[63]	Mathematical optimization	A resource management scheme and an execution model of computation and communication in PVMEC.	NP-hard problem, and resource pricing has been neglected
	[64]	prediction optimization	A dynamic multi-objective evolutionary algorithm with a multi-strategy fusion response (DMOEAD-MSFR).	NP-hard problem, and the remaining energy of PVs has been neglected.
	[65]	deep reinforcement learning(DRL)	Advanced offloading task approach through HHDL architecture for ICVs-VEC networks to maintain critical decision-making capabilities	The proposed system model contains too many assumptions, which affect the reliability of conclusions.

III. FUTURE RESEARCH DIRECTIONS

The proposed approaches in previous studies have raised open issues in FN-PV offloading schemes that can lead the future research on this topic. We can discuss them as follows:

- The parking duration time of the PVs is not taken into account in most of the previous studies. Therefore, the scholar can focus on optimizing offloading approaches that consider the parking duration time of PV with the deadline of the offloaded task.
- Most methods of the existing service migration and interruption focus on minimizing service latency while considering some feature variability, such as parking time and computing capacity across PVs. However, the energy and cost of PVs have been neglected. Therefore, considering all these metrics of PV for service migration can improve the making decision of selecting the optimal PV.
- Most of the studies assumed that the offloaded tasks are independent. Therefore, there remains a significant need to consider dependent task offloading [52] under some factors, such as network communication between FN and PVs. This problem becomes difficult to solve when the processing time of these tasks is different due to the limited resource availability of PVs.
- Most of the previous studies focus on processing one task from one end user, whereas processing multiple tasks from different end users has been neglected. In such cases, synchronous processing can be a difficult challenge. Therefore, proposing new online offloading methods for multiple tasks from multiple users can be a motivation problem to solve.
- Moreover, simulation scenarios were the traditional validation results of most proposed methods, while the tests under real-world assumptions have not been taken into account. Therefore, implementing task offloading approaches in real-world scenarios ensures their validity and applicability.
- Security and privacy concerns are overlooked in previous studies. In particular, within the FN-PV offloading environment, ensuring data integrity is

crucial, especially when tasks are offloaded to unknown parked vehicles. In addition, the proposed offloading approaches did not address secure task transmission, node authentication, or data privacy, all of which are vital in IoT systems.

- Finally, applying AI-driven offloading algorithms on different FN-PV problems is an open research area. In other words, using AI algorithms to predict the future PVs' behavior will be a very interesting research topic.

IV. CONCLUSION

The fog computing concept is closing cloud services such as computing, storage, and networking closer to the network edge. This new paradigm has enabled IoTs to offload their computation-intensive and delay-sensitive applications to FNs, in order to meet the requirements of IoT applications such as latency, cost, and energy consumption. In this context, the FN incurs a critical challenge, which is limited resources that need to be addressed to satisfy the objective of the FN and the end user. Therefore, the FN assists resources by collaborating with the idle resources of PVs. In this paper, we reviewed recent research related to task offloading approaches of the FN-PV scheme. Specifically, we have identified and classified different tasks and task offloading approaches from the perspectives of the desired objectives. In other words, we proposed a taxonomy based on popular objectives and explained their associated research issues. We also focused on a perspective model for offloading approaches in the FN-PV system and opened several research directions for further exploration of the related problems.

REFERENCES

- [1] F. Sarhaddi, Nguyen, N.T., Zuniga, A., Khan, M., Hui, P., Tarkoma, S., Flores, H. and Nurmi, P., "Llms and iot: A comprehensive survey on large language models and the internet of things," *IEEE Open Journal*, 2026, <https://doi.org/10.1109/OJCOMS.2026.011100>.
- [2] M. R. Rezaee, N. A. W. A. Hamid, M. Hussin, and Z. A. Zukarnain, "Fog offloading and task management in IoT-fog-cloud environment: Review of algorithms,

- networks, and SDN application," *IEEE Access*, vol. 12, pp. 39058-39080, 2024, <https://doi.org/10.1109/ACCESS.2024.3375368>.
- [3] M. Mukherjee *et al.*, "Latency-driven parallel task data offloading in fog computing networks for industrial applications," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 9, pp. 6050-6058, 2019, <https://doi.org/10.1109/TII.2019.2957129>.
- [4] M. Aazam, S. Zeadally, and K. A. Harras, "Offloading in fog computing for IoT: Review, enabling technologies, and research opportunities," *Future Generation Computer Systems*, vol. 87, pp. 278-289, 2018, <https://doi.org/10.1016/j.future.2018.04.057>.
- [5] X. Hou, Y. Li, M. Chen, D. Wu, D. Jin, and S. Chen, "Vehicular fog computing: A viewpoint of vehicles as the infrastructures," *IEEE transactions on vehicular technology*, vol. 65, no. 6, pp. 3860-3873, 2016, <https://doi.org/10.1109/TVT.2016.2532863>.
- [6] X. Hu, X. Tang, Y. Yu, S. Qiu, and S. Chen, "Joint load balancing and offloading optimization in multiple parked vehicle-assisted edge computing," *Wireless Communications and Mobile Computing*, vol. 2021, no. 1, p. 8943862, 2021, <https://doi.org/10.1155/2021/8943862>.
- [7] A. A. Khadir and S. A. H. Seno, "SDN-based offloading policy to reduce the delay in fog-vehicular networks," *Peer-to-Peer Networking and Applications*, vol. 14, no. 3, pp. 1261-1275, 2021, <https://doi.org/10.1007/s12083-020-01066-2>.
- [8] C. Tang, S. Xia, Q. Li, W. Chen, and W. Fang, "Resource pooling in vehicular fog computing," *Journal of Cloud Computing*, vol. 10, no. 1, p. 19, 2021, <https://doi.org/10.1186/s13677-021-00233-x>.
- [9] S.-s. Lee and S. Lee, "Resource allocation for vehicular fog computing using reinforcement learning combined with heuristic information," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 10450-10464, 2020, <https://doi.org/10.1109/JIOT.2020.2996213>.
- [10] H. Zhao, J. Hua, Z. Zhang, and J. Zhu, "Deep Reinforcement Learning-Based Task Offloading for Parked Vehicle Cooperation in Vehicular Edge Computing," *Mobile Information Systems*, vol. 2022, no. 1, p. 9218266, 2022, <https://doi.org/10.1155/2022/9218266>.
- [11] C. Ma, J. Zhu, M. Liu, H. Zhao, N. Liu, and X. Zou, "Parking edge computing: Parked-vehicle-assisted task offloading for urban VANETs," *IEEE Internet of Things Journal*, vol. 8, no. 11, pp. 9344-9358, 2021, <https://doi.org/10.1109/JIOT.2021.3056396>.
- [12] J. H. Sun, S. Choudhury, and K. Salomaa, "Multi-Objective Task Assignment Solution for Parked Vehicular Computing," in *ICORES*, 2023, pp. 193-202, doi: <https://doi.org/10.5220/0011783100003396>.
- [13] C. Li, S. Wang, X. Huang, X. Li, R. Yu, and F. Zhao, "Parked vehicular computing for energy-efficient internet of vehicles: A contract theoretic approach," *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 6079-6088, 2018, <https://doi.org/10.1109/JIOT.2018.2869892>.
- [14] A. Yousefpour *et al.*, "FOGPLAN: A lightweight QoS-aware dynamic fog service provisioning framework," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 5080-5096, 2019, <https://doi.org/10.1109/JIOT.2019.2896311>.
- [15] M. Sheikh Sofla, M. Haghi Kashani, E. Mahdipour, and R. Faghieh Mirzaee, "Towards effective offloading mechanisms in fog computing," *Multimedia Tools and Applications*, pp. 1-46, 2022, <https://doi.org/10.1007/s11042-021-11423-9>.
- [16] K. Gasmi, S. Dilek, S. Tosun, and S. Ozdemir, "A survey on computation offloading and service placement in fog computing-based IoT," *the Journal of Supercomputing*, vol. 78, no. 2, pp. 1983-2014, 2022, <https://doi.org/10.1007/s11227-021-03941-y>.
- [17] S. Misra and N. Saha, "Detour: Dynamic task offloading in software-defined fog for IoT applications," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 5, pp. 1159-1166, 2019, <https://doi.org/10.1109/JSAC.2019.2906793>.
- [18] J. Almutairi and M. Aldossary, "A novel approach for IoT tasks offloading in edge-cloud environments," *Journal of cloud computing*, vol. 10, no. 1, p. 28, 2021, <https://doi.org/10.1186/s13677-021-00243-9>.
- [19] S. Aljanabi and A. Chalechale, "Improving IoT services using a hybrid fog-cloud offloading," *IEEE Access*, vol. 9, pp. 13775-13788, 2021, <https://doi.org/10.1109/ACCESS.2021.3052458>.
- [20] S. Azizi, M. Shojafar, J. Abawajy, and R. Buyya, "Deadline-aware and energy-efficient IoT task scheduling in fog computing systems: A semi-greedy approach," *Journal of network and computer applications*, vol. 201, p. 103333, 2022, <https://doi.org/10.1016/j.jnca.2022.103333>.
- [21] C. Jiang, X. Cheng, H. Gao, X. Zhou, and J. Wan, "Toward Computation Offloading in Edge Computing: A Survey," *IEEE Access*, vol. 7, pp. 131543-131558, 2019, <https://doi.org/10.1109/ACCESS.2019.2938660>.
- [22] A. M. Alwakeel, "An overview of fog computing and edge computing security and privacy issues," *Sensors*, vol. 21, no. 24, p. 8226, 2021, <https://doi.org/10.3390/s21248226>.
- [23] E. M. T. A. Alsaadi, S. M. Fayadh, and A. Alabaichi, "A review on security challenges and approaches in the cloud computing," in *AIP Conference*

- Proceedings*, 2020, vol. 2290, no. 1: AIP Publishing LLC, p. 040022, doi: <https://doi.org/10.1063/5.0027460>.
- [24] D. Alsadie, "Artificial intelligence techniques for securing fog computing environments: trends, challenges ,and future directions," *IEEE Access*, 2024, <https://doi.org/10.1109/ACCESS.2024.3463791>.
- [25] P. Memari, S. S. Mohammadi, F. Jolai, and R. Tavakkoli-Moghaddam, "A latency-aware task scheduling algorithm for allocating virtual machines in a cost-effective and time-sensitive fog-cloud architecture," *The Journal of Supercomputing*, vol. 78, no. 1, pp. 93-122, 2022, <https://doi.org/10.1007/s11227-021-03868-4>.
- [26] A. Hazra, P. Rana, M. Adhikari, and T. Amgoth, "Fog computing for next-generation internet of things: fundamental, state-of-the-art and research challenges," *Computer Science Review*, vol. 48, p. 100549, 2023, <https://doi.org/10.1016/j.cosrev.2023.100549>.
- [27] J. Misirli and E. Casalicchio, "An Analysis of Methods and Metrics for Task Scheduling in Fog Computing," *Future Internet*, vol. 16, no. 1, p. 16, 2023, <https://doi.org/10.3390/fi16010016>.
- [28] F. Sufyan and A. Banerjee, "Computation offloading for smart devices in fog-cloud queuing system," *IETE Journal of Research*, vol. 69, no. 3, pp. 1509-152 ,1 ,2023 <https://doi.org/10.1080/03772063.2020.1870876>.
- [29] D. Alsadie, "A comprehensive review of AI techniques for resource management in fog computing: Trends, challenges and future directions," *IEEE Access*, 2024, <https://doi.org/10.1109/ACCESS.2024.3447097>.
- [30] R. Chen, Y. Fan, M. Jia, and S. Yuan, "A Game-Theoretic Scheme for Parked Vehicle-Assisted MEC Computation Offloading," *Scientific Programming*, vol. 2022, no. 1, p. 7394689, 2022, <https://doi.org/10.1155/2022/7394689>.
- [31] F. Alenizi and O. Rana, "Dynamically controlling offloading thresholds in fog systems," *sensors*, vol. 21, no. 7, p. 2512, 2021, <https://doi.org/10.3390/s21072512>.
- [32] A. J. Kadhim and J. I. Naser, "Toward electrical vehicular ad hoc networks: E-VANET," *Journal of Electrical Engineering & Technology*, vol. 16, no. 3, pp. 1667-1683, 2021, <https://doi.org/10.1007/s42835-021-00687-8>.
- [33] M. Jia, J. Zhu, and H. Huang, "Energy and delay-aware massive task scheduling in fog-cloud computing system," *Peer-to-Peer Networking and Applications*, vol. 14, no. 4, pp. 2139-2155, 2021, <https://doi.org/10.1007/s12083-021-01118-1>.
- [34] J. Xu, X. Sun, R. Zhang, H. Liang, and Q. Duan, "Fog-cloud task scheduling of energy consumption optimisation with deadline consideration," *International Journal of Internet Manufacturing and Services*, vol. 7, no. 4, pp. 375-392, 2020, <https://doi.org/10.1504/IJIMS.2020.110228>.
- [35] T. Hameed, B. Jamil, and H. Ijaz, "Efficient resource scheduling in fog: A multi-objective optimization approach," *Proceedings of the Pakistan Academy of Sciences: A. Physical and Computational Sciences*, vol. 61, no. 1, pp. 19-31, 2024, [https://doi.org/10.53560/PPASA\(60-1\)674](https://doi.org/10.53560/PPASA(60-1)674).
- [36] P. Choppa and B. Lokesh, "Efficient task scheduling and load balancing in fog computing for crucial healthcare through deep reinforcement learning," *IEEE Access*, 2025, <https://doi.org/10.1109/ACCESS.2025.3539336>.
- [37] D. Yu and W. Zheng, "A hybrid evolutionary algorithm to improve task scheduling and load balancing in fog computing," *Cluster Computing* , vol. 28, no. 1, p. 74, 2025, <https://doi.org/10.1007/s10586-024-04749-0>.
- [38] A. A. Khadir, S. A. H. Seno, B. F. Dhahir, and R. Budiarto, "Efficient-Cost Task Offloading Scheme in Fog-Internet of Vehicle Networks," *Computer Systems Science and Engineering* ,vol. 45, no. 2, pp. 2223-2234, 2023, <http://dx.doi.org/10.32604/csse.2023.032316>.
- [39] X. Huang, R. Yu, J. Liu, and L. Shu, "Parked vehicle edge computing: Exploiting opportunistic resources for distributed mobile applications," *IEEE Access*, vol. 6, pp. 66663-66649 , 2018 , <https://doi.org/10.1109/ACCESS.2018.2879578>.
- [40] C. Tang, C. Zhu, X. Wei, W. Chen, and J. J. Rodrigues, "Rsu-empowered resource pooling for task scheduling in vehicular fog computing," in *2020 International Wireless Communications and Mobile Computing (IWCMC)*, 2020: IEEE, pp. 1758-1763, doi: <https://doi.org/10.1109/IWCMC48107.2020.9148290>.
- [41] K. Nguyen, S. Drew, C. Huang, and J. Zhou, "Parked vehicles task offloading in edge computing," *IEEE Access*, vol. 10, pp. 41592-41606, 2022 , <https://doi.org/10.1109/ACCESS.2022.3167641>.
- [42] S. I. AlShathri, D. S. Hassan, and S. A. Chelloug, "Latency-aware dynamic second offloading service in SDN-based fog architecture," *Computers, Materials and Continua*, vol. 75, no. 1, pp. 1501-1526, 2023 , <https://doi.org/10.32604/cmc.2023.035602>.
- [43] M. Masdari and H. Khezri, "Efficient offloading schemes using Markovian models: a literature review," *Computing*, vol. 102, no. 7, pp. 1673-1716, 2020, <https://doi.org/10.1007/s00607-020-00812-x>.

- [44] D. H. Abdulazeez and S. K. Askar, "A novel offloading mechanism leveraging fuzzy logic and deep reinforcement learning to improve IoT application performance in a three-layer architecture within the fog-cloud environment," *IEEE Access*, vol. 12, pp. 39936-39952, 2024, <https://doi.org/10.1109/ACCESS.2024.3376670>.
- [45] K. Lone and S. A. Sofi, "A review on offloading in fog-based Internet of Things: Architecture, machine learning approaches, and open issues," *High-Confidence Computing*, vol. 3, no. 2, p. 100124, 2023, <https://doi.org/10.1016/j.hcc.2023.100124>.
- [46] M. Fahad, M. Shojafar, M. Abbas, I. Ahmed, and H. Ijaz, "A multi-queue priority-based task scheduling algorithm in fog computing environment," *Concurrency and Computation: Practice and Experience*, vol. 34, no. 28, p. e7376, 2022, <https://doi.org/10.1002/cpe.7376>.
- [47] S. Ge, M. Cheng, X. He, and X. Zhou, "A two-stage service migration algorithm in parked vehicle edge computing for internet of things," *Sensors*, vol. 20, no. 10, p. 2786, 2020, <https://doi.org/10.3390/s20102786>.
- [48] W. Fan, J. Liu, M. Hua, F. Wu, and Y. a. Liu, "Joint task offloading and resource allocation for multi-access edge computing assisted by parked and moving vehicles," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 5, pp. 5330-314, 2022, <https://doi.org/10.1109/TVT.2022.3149937>.
- [49] A. J. Kadhim and J. I. Naser, "Proactive load balancing mechanism for fog computing supported by parked vehicles in IoV-SDN," *China Communications*, vol. 18, no. 2, pp. 271-289, 2021, <https://doi.org/10.23919/JCC.2021.02.019>.
- [50] Y. Du, K. Zhang, and Y. Peng, "Interruption-Aware Task Offloading for Vehicular Edge Computing of Parking Lots," in *2023 IEEE Smart World Congress (SWC)*, 2023: IEEE, pp. 1-6, doi: <https://doi.org/10.1109/SWC57546.2023.10448987>.
- [51] C. Liu, M. Zhao, H. Wang, B. Cheng, J. Liu, and P. Yuan, "Stackelberg-game computation offloading scheme for parked vehicle-assisted VEC and experiment analysis," *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 9, pp. 5694 - 5705 , 2024, <https://doi.org/10.1109/TIV.2024.3357076>.
- [52] G. Lu, Z. Lv, Z. Zhang, and F. Zeng, "Parked Vehicles Assisted Task Offloading Based on Deep Reinforcement Learning," pp. 165-177, 2024.
- [53] Z. Ning et al., "Deep reinforcement learning for intelligent internet of vehicles: An energy-efficient computational offloading scheme," *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 4, pp. 1060-1072, 2019, <https://doi.org/10.1109/TCCN.2019.2930521>.
- [54] M. K. Farimani, S. Karimian-Aliabadi, and R. Entezari-Maleki, "Computation offloading strategy for autonomous vehicles," in *27th International Computer Conference, Computer Society of Iran (CSICC)*, 2022: IEEE, pp. 1-6, doi: <https://doi.org/10.1109/CSICC55295.2022.9780510>.
- [55] J. Zhang, X. Huang, and R. Yu, "Optimal task assignment with delay constraint for parked vehicle assisted edge computing: A stackelberg game approach," *IEEE Communications Letters*, vol. 24, no. 3, pp. 598-602, 2019, <https://doi.org/10.1109/LCOMM.2019.2961663>.
- [56] D. Han, W. Chen, and Y. Fang, "A dynamic pricing strategy for vehicle assisted mobile edge computing systems," *IEEE Wireless Communications Letters*, vol. 8, no. 2, pp. 420-423, 2018, <https://doi.org/10.1109/LWC.2018.2874635>.
- [57] C. Li, K. Xiao, J. Xu, and W. Ji, "Parked-Vehicle-Assisted Task Offloading in Vehicular Edge Computing: A Stackelberg Game Approach," *Computer Networks*, p. 111800, 2025, <https://doi.org/10.1016/j.comnet.2025.111800>.
- [58] X. Huang, R. Yu, S. Xie, and Y. Zhang, "Task-container matching game for computation offloading in vehicular edge computing and networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 10, pp. 6242-6255, 2020, <https://doi.org/10.1109/TITS.2020.2990462>.
- [59] X. Huang, R. Yu, D. Ye, L. Shu, and S. Xie, "Efficient workload allocation and user-centric utility maximization for task scheduling in collaborative vehicular edge computing," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 4, pp. 3773-3787, 2021, <https://doi.org/10.1109/TVT.2021.3064426>.
- [60] X.-Q. Pham, T. Huynh-The, E.-N. Huh, and D.-S. Kim, "Partial computation offloading in parked vehicle-assisted multi-access edge computing: A game-theoretic approach," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 9, pp. 102-20, 2022, <https://doi.org/10.1109/TVT.2022.3182378>.
- [61] X.-Q. Pham and D.-S. Kim, "Offloading Utility Optimization in Parked Vehicular Edge Computing," in *2023 14th International Conference on Information and Communication Technology Convergence (ICTC)*, 2023: IEEE, pp. 350-352, doi: <https://doi.org/10.1109/ICTC58733.2023.10392326>.
- [62] Q. Deng and F. Zeng, "Vehicular edge-computing framework for making use of parking and charging electric vehicles," *Applied Sciences*, vol. 13, no. 6, p. 4065, 2023, <https://doi.org/10.3390/app13064065>.
- [63] A. Zhou, X. Ma, S. Gao, and S. Wang, "Providing reliable service for parked-vehicle-assisted mobile edge computing," *ACM Transactions on Internet Technology*, vol. 22, no. 4, pp. 1-24, 2022, <https://doi.org/10.113514242/45>.
- [64] Y. Zhou, Z.-Y. Chai, Y.-L. Li, and J.-J. Li, "Parking Vehicle-Assisted Task Offloading in Edge

Computing: A dynamic multi-objective evolutionary algorithm with multi-strategy fusion response," *Swarm and Evolutionary Computation*, vol. 94 ,p. 101900, 2025, <https://doi.org/10.1016/j.swevo.2025.101900>.

[65] J. Liu, Y. Zou, G. Du, X. Zhang, and J. Wu, "Task Offloading and Resource Allocation for ICVs in Vehicular Edge Computing Networks Based on Hybrid Hierarchical Deep Reinforcement Learning ", *Sensors*, vol. 25, no. 22, p. 6914, 2025, <https://doi.org/10.3390/s25226914>.