Karbala International Journal of Modern Science

Volume 11 | Issue 1 Article 4

An Energy Resource Management for Cluster based IoHV Supported by Fog Computing

Ahmed Jawad Kadhim
Ministry of Education, General Directorate for Education in Al- Qadisiyah, Al- Qadisiyah, Iraq,
ahmed_al_shaibany@yahoo.com

Recommended Citation
Kadhim, Ahmed Jawad (2024) "An Energy Resource Management for Cluster based IoHV Supported by Fog Computing," *Karbala International Journal of Modern Science*: Vol. 11 : Iss. 1 , Article 4.
Available at: https://doi.org/10.33640/2405-609X.3387

An Energy Resource Management for Cluster based IoHV Supported by Fog Computing

Abstract

Internet of Hybrid Vehicle Networks (IoHV) is a network generated by merging the Internet with a Hybrid Vehicular Ad-Hoc Network (H-VANET). In IoHV, various types of electric and fuel vehicles create tasks. However, executing several tasks by electric vehicles affects their lifetime because they suffer from energy limitation issues which is one of the IoHV challenges. On the other hand, fuel vehicles and fog nodes have unlimited energy and can be used to execute most tasks of electric vehicles quickly. In this paper, we produce a new Energy Resource management Technique for IoHV called ERTH that aims to offload the tasks of electric vehicles to the fuel vehicles and fog nodes. The main goal of ERTH is saving energy and the lifetime of electric vehicles. As a result, the probability of switching off electric vehicles and the number, time, and cost of recharging their batteries will be reduced. Moreover, we propose a new energy-aware clustering method for IoHV to connect electric vehicles, fuel vehicles, and fog nodes. It can help save the energy of electric vehicles and balance the load. The results showed that ERTH is better than PLIFS and RMOIE according to EC. Moreover, NDEV with ERTH is approximately 47.2% and 42.4% with different numbers of fuel vehicles and 26.19% and 14.1% with various mobility speeds less than PLIFS and RMOIE, respectively. Finally, PET with ERTH is 2.1% and 3.09% with various numbers of fuel vehicles and 1.9% and 3.18% with various speeds more than PLIFS and RMOIE, respectively.

Keywords

H-VANET, IoHV, Electric vehicles, energy, Resource management, Fog computing

RESEARCH PAPER

An Energy Resource Management for Cluster Based IoHV Supported by Fog Computing

Ahmed J. Kadhim

Ministry of Education, General Directorate for Education in Al-Qadisiyah, Al-Qadisiyah, Iraq

Abstract

Internet of Hybrid Vehicle Networks (IoHV) is a network generated by merging the Internet with a Hybrid Vehicular Ad-Hoc Network (H-VANET). In IoHV, various types of electric and fuel vehicles create tasks. However, executing several tasks by electric vehicles affects their lifetime because they suffer from energy limitation issues which is one of the IoHV challenges. On the other hand, fuel vehicles and fog nodes have unlimited energy and can be used to execute most tasks of electric vehicles quickly. In this paper, we produce a new Energy Resource management Technique for IoHV called ERTH that aims to offload the tasks of electric vehicles to the fuel vehicles and fog nodes. The main goal of ERTH is saving energy and the lifetime of electric vehicles. As a result, the probability of switching off electric vehicles and the number, time, and cost of recharging their batteries will be reduced. Moreover, we propose a new energy-aware clustering method for IoHV to connect electric vehicles, fuel vehicles, and fog nodes. It can help save the energy of electric vehicles and balance the load. The results showed that ERTH is better than PLIFS and RMOIE according to EC. Moreover, NDEV with ERTH is approximately 47.2 % and 42.4 % with different numbers of fuel vehicles and 26.19 % and 14.1 % with various mobility speeds less than PLIFS and RMOIE, respectively. Finally, PET with ERTH is 2.1 % and 3.09 % with various numbers of fuel vehicles and 1.9 % and 3.18 % with various speeds more than PLIFS and RMOIE, respectively.

Keywords: H-VANET, IoHV, Electric vehicles, Energy, Resource management, Fog computing

1. Introduction

A few years ago, the world focused on environmentally friendly industries to reduce air pollution and oil consumption. One of these industries is electric vehicles which may change the transportation system [1]. Now, electric vehicles are used in many countries such as the United States and Japan. It is expected that the number of electric vehicles will be growing and reaching 228 million in 2030 [2]. The capabilities of these vehicles depend on the battery capacity and their lifetime is determined depending on average energy consumption [3]. Many factors such as vehicle weight, passenger weight, mobility speed, connection, and data exchange affect the energy consumption of electric vehicles [2].

H-VANET (Table 1 contains the acronyms used in this paper, arranged alphabetically) is a new version of VANET that resulted from connecting several fuel and electric vehicles to exchange information about accidents, traffic jams, etc. [4]. In H-VANET, the vehicles can be connected through V2V connection mode and with the fog nodes using V2I mode [5]. Using the Internet to connect the vehicles of H-VANET generates a new network called IoHV. In IoHV, a set of sensors is installed in the fuel and electric vehicles. The sensors of each vehicle connect with each other and with the sensors of other fuel and electric vehicles. They monitor the vehicles and environment and generate tasks that must be executed and responded to within an acceptable response time. The tasks can be executed by the vehicles or near fog computing nodes.

Fog computing is an efficient local computing framework used to execute the generated tasks by vehicle sensors. It is better than cloud computing because it is closer to the sensors and can respond

Received 20 July 2024; revised 22 November 2024; accepted 24 November 2024.
Available online 14 December 2024

E-mail address: ahmed_al_shaibany@yahoo.com.

https://doi.org/10.33640/2405-609X.3387

Table 1. List of acronyms.

Acronym	Definition
ACK	Acknowledge
CH	Cluster head
EC	Energy consumption
ERTH	Energy Resource management Technique for IoHV
FE-VANET	Fuel and Electrical Vehicular Ad hoc Network
H-VANET	Hybrid vehicular ad hoc network
ID	Identification
IoHV	Internet of hybrid vehicles network
IoV	Internet of Vehicles
NDEV	Number of dead electric vehicles
Noti_1	Notification 1.
Noti_2	Notification 2.
Noti_3	Notification 3.
NSGA	Non-dominated sorting genetic algorithm
OMNeT++ 5.0	Objective modular network test bed version 5.0
PET	The percentage of executed tasks
PLIFS	Proactive load balancing strategy for the IoV-Fog-SDN environment
RMOIE	Resource management optimization approach for the IoV-Edge system
RSU	Roadside units
SDN	Software-defined network
SUMO 0.19.0	Simulation of urban mobility version 0.19.0
SVM	Support vector machine
V2I	Vehicle-to-infrastructure
V2V	Vehicle-to-vehicle
VANET	Vehicular ad hoc network

quickly [6]. It provides a good solution to time-sensitive applications because it reduces the response time. Moreover, it produces many benefits for IoHV such as computation, storage, etc. [7]. In addition, fog computing can help reduce energy consumption and increase the electric vehicle's lifetime by executing the tasks on behalf of these vehicles.

In IoHV, there are two opposite sides. In the former, the energy is available because the battery of the fuel vehicle is recharged frequently during the vehicle's mobility while there is a shortage of energy and the battery of the electric vehicle must recharged frequently in the latter. Thus, the execution of tasks by electric vehicles affects their lifetime. To address this issue, resource management and placement of each task with a high focus on the energy parameter represent an efficient solution. This is the motivation to produce a resource management technique to exploit the available energy of fuel vehicles to execute some tasks of electric vehicles. Moreover, in this technique, the large tasks of fuel and electric vehicles that need large computations are offloaded to the fog nodes to reduce response time and the load on the fuel vehicles. However, applying the proposed technique requires an efficient communication strategy to establish connections between vehicles and fog nodes. Thus, it drives us to produce a new energy-based clustering method to select the cluster heads according to vehicle type, energy, speed, and connection to fog nodes.

Launched tasks from electric or fuel vehicles can be processed at edge computing (i.e. vehicles) or offloaded to the fog and cloud nodes depending on available resources and the task's deadline, importance, and size. To do that, several resource management, load balancing, scheduling, resource allocation, and task assignment strategies were proposed for fog computing and IoV as shown in section 3. However, they focused on enhancing the utilization of fog resources, response time, throughput, and bandwidth consumption. Moreover, some of these strategies focused on reducing the energy consumption in the fog computing nodes without paying any attention to the vehicle energy. Thus, they are inappropriate for H-VANET and IoHV because vehicle energy is an important factor in these networks.

The contributions of this paper are as follows:

- We propose a new type of vehicular network called H-VANET, which consists of connecting old-generation vehicles (i.e. fuel vehicles) with new-generation vehicles (i.e. electric vehicles). After that, we merge the Internet with H-VANET to generate IoHV.
- We produce a new IoHV-Fog architecture consisting of two layers: the fuel and electric vehicles layer and the fog nodes layer. It groups the vehicles according to a new clustering technique which selects the cluster heads according to vehicle type, connection to fog nodes, energy, and vehicle speed.
- We propose a new energy resource management technique for IoHV called ERTH. It offloads the tasks of electric vehicles to fuel vehicles and fog nodes to be run. It aims to distribute the load and save energy of the electric vehicles.

The rest of this paper is as follows: section 2 explains H-VANET, section 3 represents the related works, section 4 shows the research methodology, section 5 explains the network architecture, section 6 shows the proposed energy resource management approach, section 7 displays the simulation and results, and section 8 represents the conclusions and future works.

2. H-VANET

Recently, there has been exponential growth in the industry of electric and hybrid vehicles that use oil and electricity as energy resources [8]. Moreover, people and governments desire to buy these types of vehicles because they are environmentally friendly and save cost. Industry of such types of vehicles increased in some countries such as Japan, the United States, and China but in varying proportions. The technical and economic reports indicate that the number of electric and hybrid vehicles will increase and may reach hundreds of millions in the next few years. Normally, these vehicles and their drivers must be communicated to transfer important information about some events [2]. This communication generates a new type of vehicular network called H-VANET or FE-VANET as shown in Fig. 1.

H-VANET is a new version of VANET that resulted from the birth of electric vehicles. It connects several fuel and electric vehicles to exchange information about accidents, weather, traffic jams, etc. It can be used in many safety, environmental, convenience, and entertainment applications. In addition to VANET challenges such as un-static topology, disconnected network, security, routing, scalability, quality of service, short range, etc. [9], H-VANET suffers from a large challenge in the energy limitation of electric vehicles.

Vehicles can be connected through V2V. Moreover, they can be connected to the fog nodes beside the roads or stations during V2I [9]. Connecting the sensors of electric and fuel vehicles by the Internet generates a new version of H-VANET called IoHV. The sensors monitor the environment and generate various tasks that must be executed and answered by the vehicles, fog computing, or cloud computing.

However, H-VANET is in the first step and needs concerted efforts in different fields. It needs to be studied from different sides to produce suitable architecture, routing protocols, resource allocation techniques, load balancing methods, security approaches, and trajectory determination strategies. Moreover, appropriate hardware and connection devices are required to focus on the energy limitation parameter of electric vehicles.

3. Related works

Many researchers focused on utilizing the edge resources to execute offloaded tasks from resource-limited IoV vehicles. They presented different types of mechanisms as follows:

3.1. Resource management mechanisms

Qafzezi et al. [10] designed a new approach to manage the resources of cloud and fog computing by applying a fuzzy logic algorithm in the SDN controller to deal with vehicle data to enhance the quality of service. However, the load on the SDN controller is high due to the increasing number of arrived demands. Moreover, available computing and storage resources of fuel vehicles are not utilized efficiently. Also, it is inappropriate for electric vehicles because it does not focus on energy consumption. Yang et al. [11] investigated the problems of vehicle-to-vehicle and device-to-device connection modes in IoV. After that, they modeled the resource management problem as a decentralized reinforcement learning to increase the capacity of connection links. Moreover, they proposed a transfer—critic policy to improve the learning efficacy with the high dynamicity of the IoV environment. Li et al. [12] proposed a new dynamic resource management approach depending on the Stackelberg game model for fog computing-based IoV systems. It aims to distribute the available fog resources among IoV vehicles fairly to minimize communication delay and response time. However, it does not consider the vehicle's speed, type, and

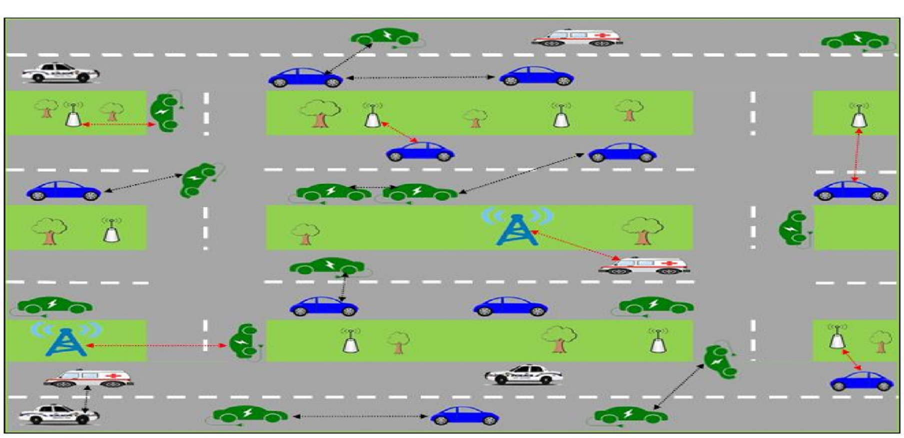

Fig. 1. Hybrid vehicular ad hoc networks (H-VANET).

energy consumption. Wang et al. [13] demonstrated that the increasing number of launched tasks from IoV vehicles leads to competition among these vehicles to get the edge resources which increases the cost, energy-wasting, and delay. Therefore, they formulated the resource management problem as a time-varying Markov decision model. Moreover, they designed a quantum-inspired reinforcement learning approach to solve this problem. Zhu et al. [14] focused on the problem of the lack of resources in edge computing and the growing number of launched tasks from IoV vehicles which increase energy consumption and delay. They proposed a resource management technique for the IoV-Fog network to reduce delay and energy waste. It depends on building two optimization models for communication and task offloading. Moreover, the authors used distributed reinforcement learning to enhance the system's efficacy and performance. However, this paper is unsuitable for IoHV because it focuses on the energy consumption of edge nodes without regard to the energy consumption of electric vehicles. Moreover, it does not utilize the available resources of fuel vehicles which can be a good assistance to computing nodes. Finally, it does not distribute the tasks among the resources fairly, and as a result, an unbalanced load will be generated.

3.2. Load balancing mechanisms

He et al. [6] merged SDN with fog computing and IoV networks and proposed a load balance mechanism based on the enhanced version of particle swarm optimization to improve the quality of service. However, it did not focus on the energy of electric vehicles. Moreover, all vehicles send fresh information frequently to the SDN manager, increasing power consumption. Kadhim et al. [15] proposed a new load-balancing technique for IoV-Fog in the SDN-based network. It collects information about the underworked and over-utilized fog nodes frequently and balances the load proactively to reduce the response time. If there is no fog node can execute a task, the parked vehicles that can be exploited to process it. However, if there are no parked vehicles or all of them are full, it computes the time of sending a task to the cloud computing servers and compares it with the expected time to that task, and according to these computations; it decides where to send this task. The problems of this paper are i. increasing the overhead and bandwidth consumption due to the frequent transfer of information even if that is no need. ii. It is not suitable for electric vehicles because it requires high energy to send information periodically. iii. It uses only the resources of parked vehicles and ignores that of moving vehicles which leads to a high number of idle resources. X. Hu et al. [16] proposed a paradigm to utilize multiple parked vehicles as assistance computing nodes to reduce the load on edge computing devices. After that, they modeled the problem of load balancing and task offloading to reduce the cost by considering the time constraint. They proposed an approach to select the favorable parked vehicles to assist edge computing in executing the offloaded tasks. Finally, they used dynamic game theory to solve the problem of task allocation to enhance efficiency and performance. Li et al. [17] proposed a hierarchical and partitioned SDN framework for the IoV network. Moreover, they suggested a load-balancing approach to schedule the tasks locally during the domain of each fog server and the tasks of all fog servers globally. Moreover, they designed a Deep Q-Network policy to solve this problem and balance the load among the available fog devices. Du et al. [18] designed a new load-balancing technique for IoV with coordinated multi-point communications. It depends on the prediction concept of the number of reached vehicles to the domain of each edge node to determine which node is under-loaded and which one is overloaded to distribute the load among them. Darade et al. [19] utilized the benefits of the SDN concept and its global knowledge to balance the unfairly distributed requests and reduce the delay in the IoV-Fog-Cloud network. Moreover, they used an integrated whale optimization method to distribute the generated demands from IoV vehicles between cloud and fog servers.

3.3. Resource allocation mechanisms

He et al. [20] proposed a non-orthogonal multiple access-based resource allocation approach for IoV to improve the total sum rate. It consists of two stages. The first one is user collecting which depends on temporal difference error-based prioritized sampling technique. The second one is resource allocation. Kalsoom et al. [21] focused on the increasing number of vehicles in the IoV network and demonstrated that this network is unfavorable for cooperative driving systems due to limited resources. Thus, they produced a new IoV architecture based on 5G networks and a device-to-device-based resource allocation approach. Moreover, they merged noise clustering and density-based scatted clustering methods to enhance the quality of service. Y. Zhang et al. [22] studied the effect of the high velocity of IoV vehicles on the complexity of task offloading and resource allocation. Then, they design a mathematical model to offload the tasks and allocate the available resources at edge computing to execute the offloaded tasks. After that, they utilized the deep reinforcement learning approach to solve that model to reduce the total cost, overhead, and latency. Hazarika et al. [23] proposed a new policy that depends on collecting information from the IoV vehicle about available services and resources to offload the tasks. Moreover, they presented a deep reinforcement learning method to categorize the tasks based on their computation size and priority to optimize the energy allocation operation. Xi et al. [24] presented a new one that depends on the deep learning network model to solve the resource allocation problem and increase the quality of service.

3.4. Tasks assignment mechanisms

Ni et al. [25] proposed a deployment technique to determine the number and location of RSUs in 2-D IoV scenarios to increase their utilization. Moreover, they proposed a task assignment strategy to decrease the delay. Zhou et al. [26] focused on unfair resource utilization and delay problems in the vehicular fog computing network. Thus, they proposed a novel contract theoretical model-based incentive mechanism to increase resource utilization. Moreover, they modeled the task assignment issue and optimized it using a matching approach based on a pricing algorithm. Peng et al. [27] focused on utilizing the embedded resources of the parked vehicles to assist the edge servers. In addition, they proposed a task assignment optimization policy based on the ternary search-based method and Stackelberg game system to increase the parked vehicles' resource utilization and reduce the system cost. Nguyen et al. [28] proposed a task assignment approach for high-mobility vehicles in highway scenarios based on distributing launched tasks from vehicles to several agents. The proposed approach used the learning concept to deal high volume of tasks and dynamic features of vehicular networks. Wu et al. [29] designed a task assignment technique based on information related to vehicle mobility speed and accuracy. They formulated and solved the task assignment problem using a matching optimization model and 0—1 integer linear programming.

3.5. Scheduling mechanisms

Pang et al. [30] designed a new collaborative Double Deep Q-Network-based scheduling technique for processing resources of IoV vehicles. Moreover, they divided the geographical into a set of areas and assumed that there is a server in each one to execute the offloaded tasks from IoV vehicles. The proposed technique aims to reduce the total system computation cost and takes into account location privacy protection. Chen et al. [31] designed an efficient resource scheduling mechanism based on game theory and blockchain for IoV considering the resource limitation and high mobility of vehicles. The tasks are launched from resource-limited vehicles to the under-utilized edge and cloud resources to be executed to enhance network efficiency and performance. Hu et al. [32] introduced a real-time scheduling framework for the IoV-Edge network that considers the latency. First, the users send their tasks to this framework which considers the vehicles as computing resources. To assign them to suitable resources, they are scheduled based on a greedy algorithm that reduces the latency and increases the full utility of resources. Feng et al. [33] presented a solution for the scheduling problem of generated tasks from mobile vehicles toward RSUs. It depends on transforming the scheduling issue into a multi-objective optimization problem and then solving it using a reliability condition-based scheduling approach to improve the cost and processing time. Awada et al. [34] investigated the resource-aware offloading problem in the IoV-edge environment. Thus, they proposed a scheme to keep the resources of vehicles and edge devices in one pool taking into consideration the capacity and vehicle mobility. Moreover, they proposed task scheduling based on a variant Bin-Packing optimization approach to maximize resource utilization.

3.6. Energy-aware mechanisms

Zhai et al. [35] proposed a new heuristic optimization-based offloading policy for the IoV-fog-SDN system. It focuses on exploiting the available energy of vehicles and the application dependence concept to run more applications with minimum response time. Nonetheless, the frequent exchange of information to the SDN controller increases energy consumption. Chen et al. [36] produced a non-orthogonal multiple access system to establish the vehicles’ communications with each other in the IoV network. After that, they utilized the proposed system to produce a new model to allocate the available resources to increase energy efficiency by producing a deep reinforcement learning-based sub-channel and energy allocation technique. Chen et al. [37] presented a new approach to allocate the resources and collect the data in IoV based on determining the age and freshness of data. In the approach, the authors partitioned the problem into a set of minimum problems to enhance the energy consumption and response time. The problems of this approach are latency, scalability, and lack of stability. Han et al. [38] designed a new decentralized resource allocation approach that depends on deep Q-network. It aims to determine the resource block to reduce the latency, power consumption, and interference and increase the sum rate of vehicle and infrastructure links. However, the proposed approach did not focus on task features such as type, size, and deadline. Miao et al. [39] proposed an energy-based optimization technique to improve the task allocation process in the Internet of Things. It uses the krill herd algorithm to allocate tasks with high levels of stability and reliability and low energy consumption. Nonetheless, the proposed technique did not take the type and importance degree of tasks into consideration. Vijarania et al. [40] proposed a load-balancing scheme to distribute the load of the Internet of Things devices among the fog nodes and cloud servers with efficient consumption of energy. However, this scheme focused on the fixed nodes and did not pay any attention to the mobility which may destroy the system performance. Khoobkar et al. [41] focused on the energy and delay when offloading the demands in a fog computing system and proposed a dual-population approach depending on using replicator dynamics and considering the interaction between fog nodes and the user. Pakmehr et al. [43] produced an efficient scheduling mechanism in fog computing based on energy and deadline factors using SVM. It uses the prediction principle to determine the traffic load and divides it into high and low traffic. Then, it schedules the traffic using NSGA and reinforcement learning. However, the proposed mechanism is unsuitable for electric vehicles because it works at the fog computing layer and does not focus on the vehicle energy limitation.

Most of the above previous research papers produced solutions to the problem of the increasing number of generated demands from vehicles and resource limitations in fog computing and IoV. The IoV vehicles are fuel and their energy is rechargeable during mobility. Thus, the energy is not limited in these types of networks and this is the answer to the question of why the researchers of the above-related works did not focus on the energy parameter in these networks and did not give it high importance as compared to other parameters such as delay, bandwidth, throughput, and to name a few. Moreover, some of the above-related works did not give any attention to the task characteristics such as type, length, deadline, etc. while scheduling the tasks and balancing the load. In addition, even the related works [35—42] focused on enhancing the energy consumption in the edge layer without focusing on the energy consumption of electric vehicles. However, with the growth in the electric vehicles industry, IoV will be replaced with IoHV. As a result, new challenges occur and need to be overcome like limitation of energy and computation and storage resources in electric vehicles. However, the energy limitation of electric vehicles is at the front of these challenges. Many sides affect the energy consumption and electric vehicles' lifetime such as inefficient techniques for resource management, load balancing, routing, and network architecture. Thus, the above mechanisms are unsuitable for H-VANET and IoHV because they do not consider the energy parameter in operations of task execution, distribution of load, routing the data, building the network, etc. This analysis is the motivation to produce H-VANET and energy-efficient resource management to solve the problem of energy limitation in electric vehicles.

4. Research methodology

The methodology of the proposed system consists of three stages as follows:

- IoHV network setup: in this stage, the vehicles are grouped in clusters and the cluster heads and gateway node(s) are determined according to a new clustering approach which depends on the vehicle type, speed, energy, and number of connected fog nodes. It aims to save the energy of electric vehicles and prevent electric vehicles with low residual energy levels from doing the roles of cluster heads.
- Resource management technique: it aims to exploit the available resources in the fuel vehicles and fog nodes to execute the tasks of electric vehicles and as a result, save the energy of electric vehicles. In this technique, the tasks are executed and offloaded to other nodes depending on the task size and importance degree and available fog nodes and fuel vehicles using two proposed algorithms.
- Simulation and analysis: in this stage, the efficacy and performance of the proposed mechanism are investigated and compared with two of the recent resource management approaches which are RMOIE [14] and PLIFS [15]. Thus, two simulation scenarios are built using OMNeT++ and SUMO simulators. These scenarios contain different numbers of fuel vehicles 1200, 1400, 1600, 1800, and 2000 vehicles moving at speeds

ranging from 10 to 20 m/s and numbers of tasks 150, 300, 450, 600, and 750 tasks with different importance degrees and volumes ranging from 50 to 300 MB. The number of fog nodes is 100. Moreover, in these scenarios, we use different residual energy levels of electric vehicles. The analysis of simulation results of the number of dead electric vehicles, energy consumption, and percentage of executed tasks by the fog nodes and fuel and electric vehicles is done based on the proposed relationships and formulas.

5. System architecture

System architecture as shown in Fig. 2 composes of two layers: fog computing and vehicles. The fog computing layer consists of several RSUs that are located beside the roads and a number of routers and servers that are found inside homes and restaurants and have connections with some vehicles. The fog nodes have computing and storage capabilities to work as simple cloud computing. In the vehicles layer, the vehicles are fuel and electric.

They are grouped in clusters according to vehicle type, speed, energy, and connecting to fog nodes. To reduce the overhead, we assumed that the maximum number of fuel and electric vehicles in each cluster must be less than or equal to 20 vehicles in addition to a cluster head.

The other assumptions of the proposed system are as follows:

- The number of fuel vehicles is more than that of electric vehicles.
- The fog nodes are found in some geographical areas while no one in the other.
- The fuel vehicles and fog nodes have heterogeneous capabilities of storage and processing.
- Fuel and electric vehicles connect with each other and fog nodes using a WiFi connection.
- There is no connection among the fog nodes.
- The electric and fuel vehicles move at various speeds.
- The levels of residual energy in the electric vehicles are not equal.
- The *energy_threshold* of each electric vehicle is equal to 15 % of its battery capacity.

To construct a cluster, there is a need to determine its cluster head to do the controlling and managing operations. However, fuel vehicle has a higher priority to work as a cluster head than electric vehicle. Moreover, the fuel vehicle connecting to a fog node has the highest priority to work as a cluster head as compared to that with no connection. To select the cluster head for each set of vehicles located in a geographical area, each one sends a HELLO message containing details about its ID, type (i.e. fuel or electric), remaining energy (only electric vehicles), speed, and number of connected fog nodes. In this article, there are two cases to select a cluster head as follows:

5.1. Case 1: fuel vehicle as cluster head

The selection of optimal cluster head depends on the number of fuel vehicles, number of connected fog nodes, and mobility speeds as follows:

- If there is only one fuel vehicle, it works as a cluster head whether it has a connection to a fog node or not (see Fig. 3).
- Otherwise (i.e. there is more than one fuel vehicle and all of them have no connection with

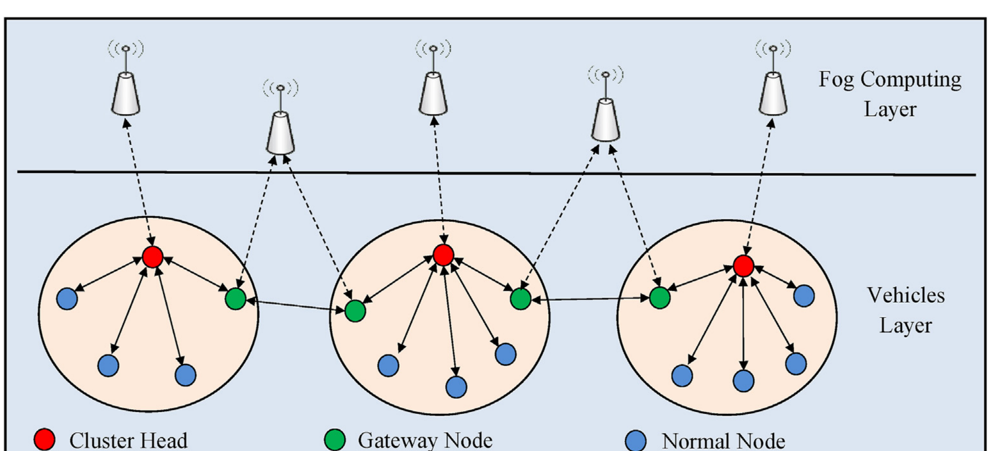

Fig. 2. System architecture.

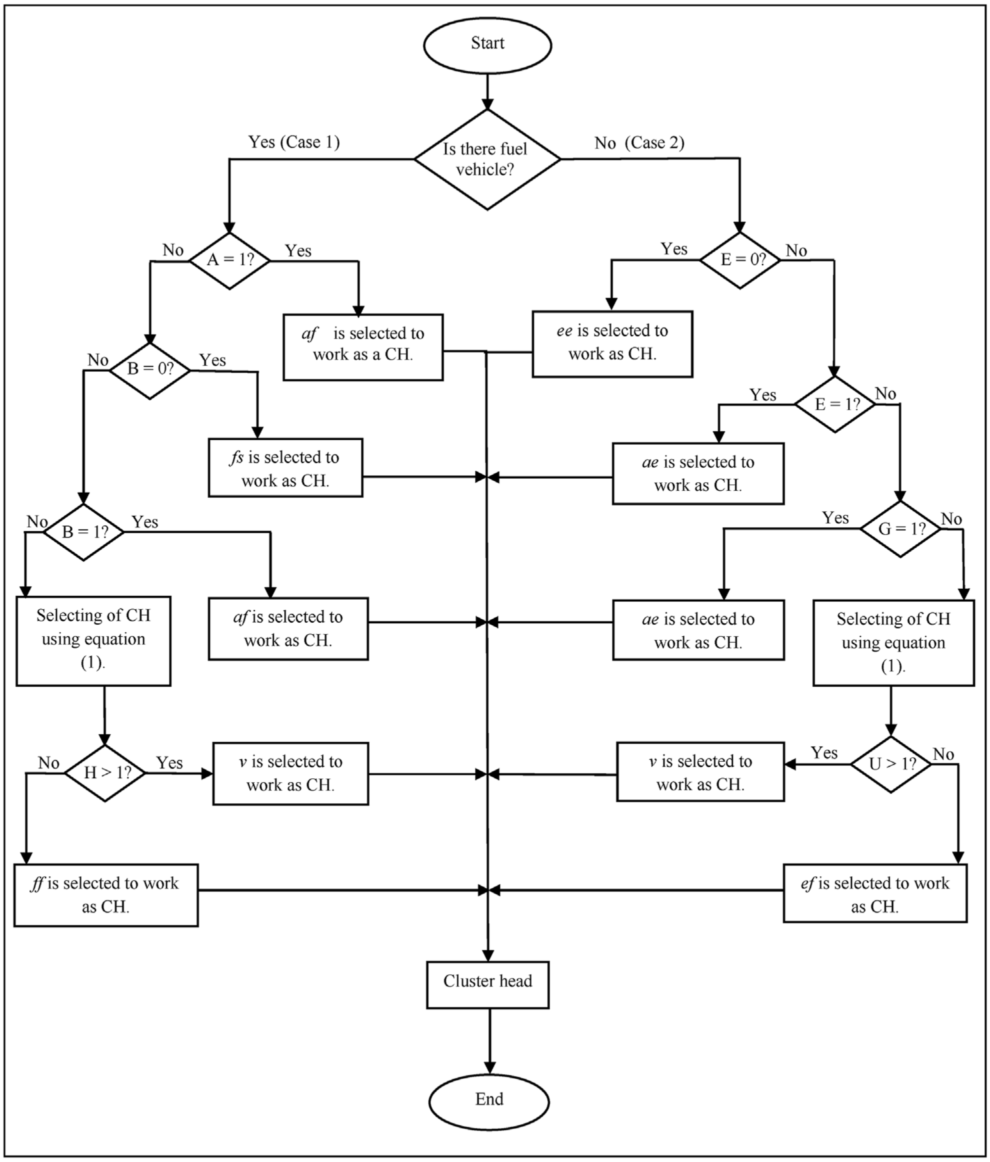

Fig. 3. Strategy of selecting optimal cluster head.

fog nodes, the vehicle with minimum speed is selected to work as a cluster head. Otherwise, if only one of them has a connection to fog node(s), it will work as a cluster head. Otherwise (i.e. there is more than one fuel vehicle with a connection to fog node(s), the selection operation of cluster head is done based on the number of connected fog nodes and mobility speed of the fuel vehicle). The fuel vehicle with minimum mobility speed and the maximum number of connected fog nodes has the highest priority to play as a cluster head. To select the best fuel vehicle to work as cluster head, a fitness value is computed as follows:

$$Min\, Fitness(i) = S_i - K_i,\ \forall i \in C \tag{1}$$

S_i and K_i represent the normalized values of the mobility speed of fuel vehicle i and the number of connected fog nodes to fuel vehicle i respectively. See Table 2 for other notations used in this paper. The values of S_i and K_i are computed by dividing the mobility speed of vehicle i (s_i) and the number of connected fog nodes to vehicle i (k_i) by the maximum value of mobility speed and the maximum number of connected fog nodes as follows:

$$S_i = s_i / S,\ \forall i \in C, \tag{2}$$

Table 2. Notations.

Variable	Description
A	The number of candidate fuel vehicles to work as cluster heads.
af	The available fuel vehicle.
B	The number of candidate vehicles to work as cluster heads and have connections with fog nodes.
fs	The vehicle with minimum speed.
H	The number of fuel vehicles with the same minimum fitness value.
v	The vehicle (fuel or electric) with the maximum number of connected fog nodes.
ff	The fuel vehicle with minimum fitness value.
i, C	Index, set of candidate fuel vehicles to work as cluster heads. The vehicles in *C* may be fuel (Case 1) or electric (Case 2).
s_i	The current mobility speed of fuel vehicle *i*. Its value before normalization.
S	Maximum speed. The speeds of all candidate vehicles to work as cluster heads are compared with each other and the maximum one is selected using Equation (4).
k_i	The current number of connected fog nodes to fuel vehicle *i*. Its value before normalization.
K	Maximum number of fog nodes. The numbers of connected fog nodes of all candidate vehicles to work as cluster heads are compared with each other and the maximum one is selected using equation (5).
E	The number of electric vehicles with more remaining energy than *energy_threshold*.
ee	The electric vehicle with the highest remaining energy.
ae	The available electric vehicle.
G	The number of electric vehicles with more remaining energy than *energy_threshold* and have connections to the fog node.
U	The number of electric vehicles with the same minimum fitness value.
ef	The electric vehicle with minimum fitness value.
Noti_1	Notification 1. The value (Noti_1 = 1) means that the vehicle received notification 1.
Noti_2	Notification 2. The value (Noti_2 = 1) means that the vehicle received notification 2.
Z_i	The total number of fuel vehicles found in the domain of cluster head *i* and the number of 3 node(s) that connected to cluster head and gateway nodes of cluster *i*.
X_i	The number of fuel vehicles that are found in the domain of cluster head *i*.
Y_i	The number of fog node(s) that connected to cluster head and gateway node(s) of cluster *i*.
AC	Set of available cluster heads for the new incoming vehicle.
E_i	The total number of electric vehicles found in the domain of cluster head *i*.
D_i	The difference value of cluster head *i*.
t, $size_t$	A task, the size of task *t*.
λ	The threshold of allowed size by the vehicle.
β_i	Set of available fog nodes for cluster head *i*. These fog nodes may connect to the cluster head *i* directly or to the gateway nodes that are found in the cluster of cluster head *i*.
f_j, M	Fog node *j*, Set of all fog nodes.
c_{ij}	A connection between cluster-head *i* and fog node *j*. If it is value is 1, the connection is available. Else, there is no connection.
C_{qj}	A connection between gateway *q* and fog node *j*. If it is value is 1, the connection is available. Else, there is no connection.
q, G_i	The gateway node, A set of gateway nodes that belongs to the cluster head *i*.
CH, N	Cluster Head, Set of all cluster heads.
f, e	A fuel vehicle, An electric vehicle.
Q_f	The queue of fuel vehicle *f*.
R_i	The number of available (i.e. under-loaded) fog nodes for cluster head *i*.
F	The fog node with minimum load.
L_i	The load of fog node *i*.
I_i	Importance of task *i*.
W	The waiting queue of the electric vehicle.
p	A waiting time of the task in the waiting queue before receiving a notification.
V_i	The number of available (i.e. under-loaded) fuel vehicles in the domain of cluster head *i*.
Υ_i, D	The load of fuel vehicle *i*, The fuel vehicle with minimum load.

$$K_i = k_i / K, \ \forall i \in C, \quad (3)$$

where

$$S = \underset{i \in C}{Max}\, S_i, \quad (4)$$

$$K = \underset{i \in C}{Max}\, K_i, \quad (5)$$

The values of mobility speed and number of connected fog nodes are opposites. The fuel vehicle with minimum mobility speed is the best one while the fuel vehicle with the maximum number of connected fog nodes is the best. Thus, in (1), we use a minimum objective function and multiply the value of K_i by a negative sign.

If there is more than one vehicle with the same best (i.e. minimum) fitness value, one of these vehicles which has the maximum number of connected fog nodes will be cluster head to increase the available resources.

The following example is to explain how to select an appropriate cluster head using equation (1). In this example, there are 3 fuel vehicles (1, 2, and 3) that move at different speeds and have varying numbers of connected fog nodes. Vehicles 1, 2, and 3 move at 20 m/s, 15 m/s, and 15 m/s, respectively, and have 3, 3, and 2 connected fog nodes, respectively. After applying equations (4) and (5), the values of S and K are 20 m/s and 3 fog nodes, respectively. To select the best one from these three vehicles, we must compute the values of S_i and K_i using equations (2) and (3) and the fitness value of each vehicle using equation (1) as follows:

$S_1 = 20/20 = 1$, $K_1 = 3/3 = 1$,

$\text{Fitness}_1 = 1 - 1 = 0$

$S_2 = 15/20 = 0.75$, $K_2 = 3/3 = 1$,

$\text{Fitness}_2 = 0.75 - 1 = \mathbf{-0.25}$

$S_3 = 15/20 = 0.75$, $K_3 = 2/3 = 0.66$,

$\text{Fitness}_3 = 0.75 - 0.66 = 0.09$

In this example, fuel vehicle 2 has the minimum fitness value and it is the best one to work as a cluster head.

5.2. Case 2: electric vehicle as cluster head

If there is no fuel vehicle (i.e. all vehicles are electric), one of the electric vehicles will work as a cluster head. See Fig. 3. Of course, determining the optimal cluster head depends primarily on the remaining energy in the electric vehicles as follows:

- If there is no electric vehicle with remaining energy more than *energy_threshold* (i.e. all the electric vehicles have little energy), the electric vehicle that has the highest remaining energy among all of them is selected to play the role of cluster head. Of course, the electric vehicle that has the highest remaining energy and connection(s) to the fog node(s) takes the maximum priority to work as a cluster head.
- If there is only one electric vehicle with remaining energy more than *energy_threshold*, it is selected to work as cluster head.
- If there is more than one electric vehicle with remaining energy more than *energy_threshold*, the vehicle that has a connection to a fog node, will work as a cluster. If there is more than one electric vehicle with remaining energy more than *energy_threshold* and has connections to fog node(s), the election of optimal cluster head is performed using equation (1).

After that, each cluster head sends an *ACK* message to all its members. The fuel vehicle(s) and connected fog node(s) send information about their load to the cluster head periodically to determine the overloading and under-loading status using the thresholds that are used in Ref. [15]. Moreover, each cluster head that has no connection to a fog node(s) during itself or its gateway node(s) or the connected fog node(s) is overloaded sends a notification called *Noti_1* to its members. Each cluster head that has no connection to a fog node during itself or its gateway node(s) and no fuel vehicle in its domain, sends a notification called *Noti_2* to all its members. Moreover, it sends *Noti-2* if the connected fog node(s) and fuel vehicle(s) that are found in its domain are overloaded. This prevents the vehicles from sending their tasks to the cluster head to find a fuel vehicle or fog node to execute their tasks even though it has no available fuel vehicle or fog node in its domain. It can reduce the overhead on the network links and cluster head and additional energy waste from the electric vehicles.

During vehicle mobility, many changes may occur such as connecting or disconnecting a cluster head or gateway node to/from a fog node and entering or exiting a fuel vehicle to/from a cluster. In addition, due to the continuous generation of new tasks and continuous execution of existing tasks, other changes may occur which are overloading or/and under-loading of fuel vehicles and connected fog node(s). According to these changes, the cluster head sends *Noti_1, Noti_2, or Noti_3* to its members. *Noti_3* is the opposite of *Noti_2* and is sent when all its members in a cluster are electric vehicles and a new fuel vehicle enters the cluster or the cluster head connects with a new under-loaded fog node. Also, the cluster head sends *Noti_3* to its members if the status of fuel vehicle(s) and/or connected fog node(s) changes from overloaded to under-loaded.

If a cluster head goes out from its cluster and the connection with its members and gateways breaks, a new cluster head from the members and gateways must be selected. To do that, the above procedure of selecting the optimal cluster head during case 1 or case 2 is repeated.

To join a new incoming vehicle to a suitable cluster, a new joining procedure to increase the available resources and their utilization is proposed. The type of incoming vehicle is taken into account in this procedure. First, the vehicle broadcasts a *JOINING* message with information about its type. Each cluster head that receives this message will check its members. If its members are less than 20, it computes the number of fuel vehicles in its domain and the number of connected fog nodes whether through itself or its gateway node(s) using equation (6). As it is known, the capabilities of the fog node are better than those of the fuel vehicle. Thus, we multiply the number of fog nodes Yi by 2 in (6). After that, if the type of the incoming vehicle is electric, the cluster head computes the difference value D_i between Z_i and E_i using equation (7). But if the vehicle type is fuel, the cluster head computes the difference value D_i between E_i and Z_i using equation (8). Finally, it responds with an *ACK* message containing information about the value of D_i.

$$Z_i = X_i + 2(Y_i), \tag{6}$$

$$D_i = Z_i - E_i \tag{7}$$

$$D_i = E_i - Z_i \tag{8}$$

In the second stage, if the incoming vehicle receives one *ACK* message from a cluster head, it joins directly to that cluster head. But if it received more than one *ACK* message from various cluster heads, the vehicle selects a suitable cluster head that has the maximum value of D_i using equation (9) as follows:

$$\underset{i \in C}{Max}\, D_i \tag{9}$$

Example: this example explains how the incoming vehicle is joined to the appropriate cluster. Let's assume that the incoming vehicle is electric and 3 cluster heads received the *JOINING* message from the incoming vehicle. Also, let's assume that the values of Z_i and E_i of each cluster head are shown in Table 3. The optimal cluster for the electric incoming vehicle is number 3.

Now, we repeat the above example but here we assume that the incoming vehicle is fuel. The optimal cluster for the fuel incoming vehicle is number 1 when $D_i = 2$. See Table 4.

Table 3. Cluster head computation when the incoming vehicle is electric.

Cluster Head	Z_i	E_i	D_i (using equation (7))
1	6	8	−2
2	6	5	1
3	5	2	3

Table 4. Cluster head computation when the incoming vehicle is fuel.

Cluster Head	Z_i	E_i	D_i (using equation (8))
1	6	8	2
2	6	5	−1
3	5	2	−3

6. Proposed energy resource management technique

Before describing the proposed technique, we must explain the benefits of the proposed system and technique on traffic management, urban planning, or environmental impact. The proposed system can help in enhancing traffic management. In this system, the fuel and electric vehicles generate tasks related to traffic jams, air conditions, vehicle breakdowns, potential accidents, sudden stops of neighbor vehicles, etc. Executing these tasks within a minimum response time is one of the main goals of the proposed system. Thus, it can help the drivers to take the optimal actions to reduce the disasters. Moreover, the proposed system has other benefits related to urban planning and the drivers of electric vehicles which are reducing the probability of vehicle turn-off and the number, cost, and time of recharging operations of electric vehicles and as a result, reducing the number of recharge stations in the cities. In addition, in the proposed technique, the fuel vehicle exploits the energy of its battery which is recharged frequently during the vehicle's mobility to perform the calculation and processing of tasks. Moreover, they execute the small tasks while the large tasks are executed by the fog nodes. Thus, the fuel consumption and CO2 emissions related to the processing operations are very little or imperceptible.

The energy of the electric vehicles is limited and depends on the battery capacity which must be recharged frequently. The cost and time of recharging depend on the energy consumption of the vehicles. On the other side, the energy of the battery of the fuel vehicles is recharged frequently through the vehicle's mobility and the energy of the fog nodes is available at all time. Therefore, the energy of the electric vehicles takes the highest priority and must be considered in the proposed systems.

However, executing the tasks by electric vehicles consumes high energy which is a very important factor for these vehicles. With the presence of replacing energy resources such as fuel vehicles and fog nodes, it is better to exploit them. Thus, the tasks of electric vehicles are offloaded to the replacing resources to reduce the energy consumption in electric vehicles. The tasks and workload are distributed so that the fuel vehicles, fog nodes, and even electric vehicles participate in executing the launched tasks. It means that the energy of the fog nodes and fuel vehicles is taken into account but with lower priority as compared with that of electric vehicles. It is the main purpose of the proposed energy resource management technique (ERTH). It aims to select the suitable resource to execute each task. In this technique, the connected fog nodes and fuel vehicles send information about the load and number of tasks in their queue to the cluster head periodically. The cluster head saves this information in its database to make a decision in the future.

The task can be processed by fog computing nodes or at the network edge by fuel and electric vehicles. The decision to execute the task at the network edge or offload it to the fog layer depends on the load and task volume. To determine whether the task is small or large, a threshold λ is assumed in this paper. See equation (10).

$$t = \begin{cases} \text{small}, \ \text{size}_t \le \lambda \\ \text{large}, \text{otherwise} \end{cases} \tag{10}$$

If the task is small, it is better to be executed at the network edge by the fuel vehicles to decrease the response time and give a chance to execute the tasks of electric vehicles. Of course, electric vehicles execute the tasks in some cases as we will explain later in Algorithm 2. The advantages of executing the small tasks at the network edge (i.e. vehicles) are i. reducing the response time due to the small distance between them. ii. Reducing the load on the fog nodes. Moreover, this reduces the load on fuel vehicles and therefore, increases the chances of electric vehicle tasks being executed by fuel vehicles. iii. Reducing the effect of vehicle mobility and receiving the response quickly before exiting the vehicle from the cluster. On the other hand, the large volume tasks of fuel and electric vehicles are offloaded to the fog nodes. The advantages of executing the large tasks at the fog nodes are: i. the fog nodes have additional computation capabilities and can respond with quick response time compared to the vehicles. ii. They can execute large tasks with higher stability than vehicles.

When a fuel vehicle *f* has a task *t*, it checks the task size. See Algorithm 1 and Fig. 4. If $size_t$ is less than or equal to λ, it enters t into its queue for processing. Otherwise, (i.e. if $size_t$ is more than λ), f checks its queue, and if it is empty, f adds t to its queue for processing. Otherwise (i.e. the task size is large and the queue is not empty), f checks the received notifications, if it received *Noti_1* or *Noti_2*, f enters t into its queue and processes it. Otherwise (i.e. there is no received notification), f forwards a request that contains information about the vehicle's ID and type and task's size to the cluster head which determines which one from the available fog nodes must execute this task depending on load. After that, the task will be sent to the selected fog node which may be connected to a cluster head or gateway node. However, if there is only one fog node available for the cluster head, f forwards the task to this fog node. The available fog nodes for each cluster head are the fog nodes that have connections with the cluster head and gateway nodes of that cluster. To determine the fog nodes that have connections with the gateway nodes, the gateways send notifications to the cluster head and inform it about the connected fog nodes and their capabilities. The available fog nodes for each cluster head are determined as follows:

$$\beta_i = \left\{ f_j \middle| C_{ij} = 1 \middle| \right\} \cup \left\{ f_j \middle| C_{q_j} = 1 \middle| \right\}, \forall i \in N, j \in M, q \in G_i \tag{11}$$

$$C_{ij} = \begin{cases} 1, \text{if there is a connection between the cluster head } i \text{ and fog node } j. \\ 0, \text{otherwise}. \end{cases} \tag{12}$$

$$C_{q_j} = \begin{cases} 1, \text{if there is a connection between the gateway node } q \text{ and fog node } j. \\ 0, \text{otherwise}. \end{cases} \tag{13}$$

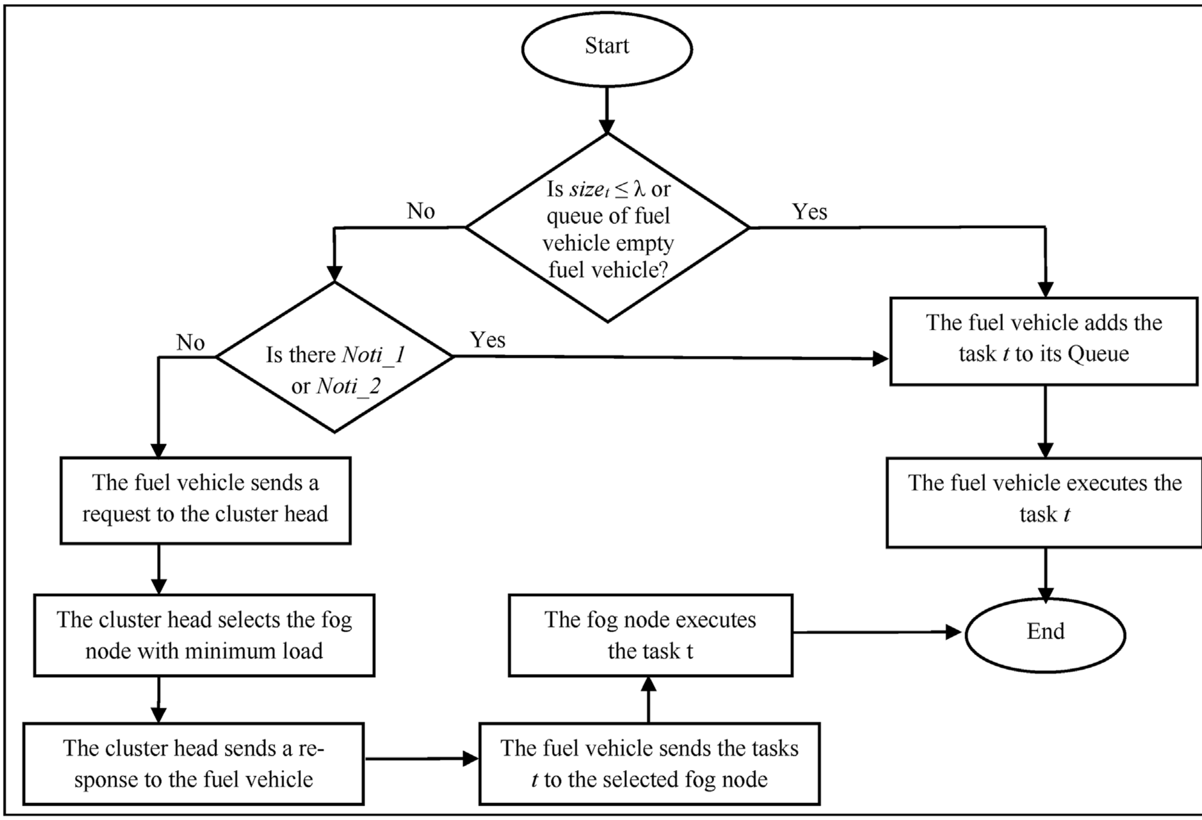

Fig. 4. Strategy of selecting the suitable vehicle or fog node to execute a task t of a fuel vehicle.

If an electric vehicle e has a task t, first it checks the received notifications, and if there is *Noti_2*, it checks the task importance using equation (14). See Algorithm 2 and Fig. 5. If its importance is high, e runs the task. Otherwise, e adds the task to the waiting queue. In this queue, the tasks stay for some time p and wait for changes in the cluster. If e receives *Noti_3*, it forwards a request that contains information about the vehicle ID and task's size and type to the cluster head. Otherwise, the electric vehicle is forced to execute the task t.

The time complexity of Algorithm 1 is O(1) and the time complexity of Algorithm 2 is O(1).

$$I_i = \begin{cases} 1, \text{if task } i \text{ is high importance.} \\ 0, \text{if task } i \text{ is low importance.} \end{cases} \quad (14)$$

Otherwise (i.e. if there is no *Noti_2*), e sends a request to the cluster head which checks the task size and the available resources as follows:

- If the task is small and there is fuel vehicle(s) in the cluster head domain or the task is large and there is no available fog node(s) connected to the cluster head, the task is executed by a fuel vehicle. If there is only one fuel vehicle, the cluster head sends a response to the electric vehicle to inform it to forward its task to this fuel vehicle. Otherwise, if there is more than one fuel vehicle, it balances the load and selects one of them that has minimum load and sends a response to the electric vehicle to send its task to the selected fuel vehicle.
- Otherwise, the task is executed by a fog node. If there is only one fog node, the cluster head informs the electric vehicle to send its task to this fog node. Otherwise, if there is more than one fog node, it balances the load and selects one of them that has minimum load and sends a response to the electric vehicle to send its task to the selected fog node.

7. Simulation and results

In this section, the goal is to prove that the recent resource management approaches of IoV are

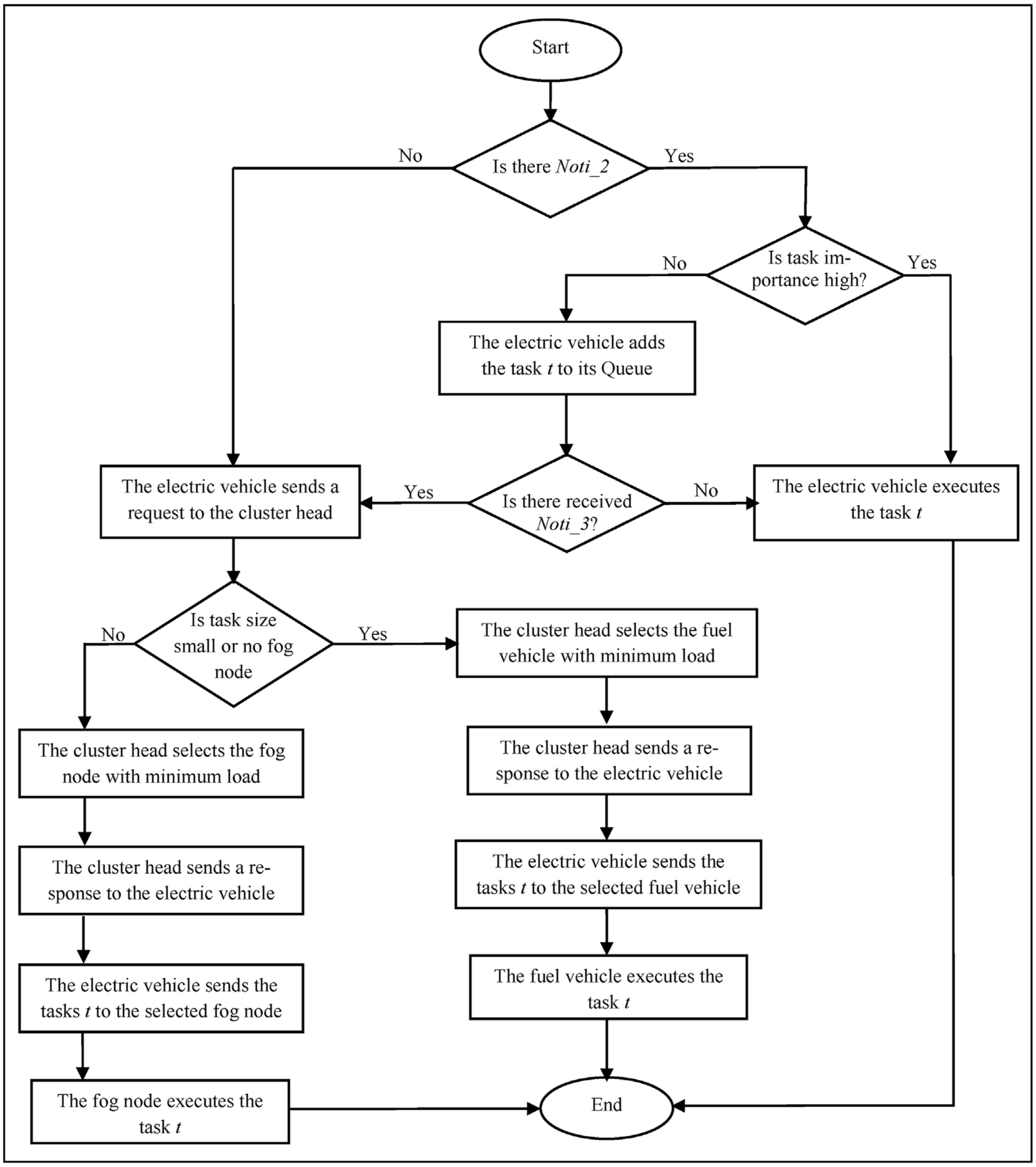

Fig. 5. Strategy of selecting the suitable vehicle or fog node to execute a task t of an electric vehicle.

unsuitable for IoHV. It can be done by comparing the resource management optimization approach for the IoV-Edge system (For simplicity, we call it RMOIE in this paper) [14] and the proactive load balancing strategy for the IoV-Fog-SDN environment (For simplicity, we call it PLIFS in this paper) [15] with the proposed technique ERTH. See Table 5 for more details about ERTH, PLIFS, and RMOIE. To measure the performance of these strategies, the following performance metrics are used:

Table 5. Comparison of ERTH, RMOIE, and PLIFS.

Parameters	RMOIE	PLIFS	ERTH
Architecture	Traditional IoV	SDN-based IoV	Traditional IoHV
Clustering concept	No	No	Yes
Fog computing	Yes	Yes	Yes
IoV vehicle Types	Fuel	Fuel	Fuel and electric
Exploiting of parked IoV vehicles	No	Yes	No
Major Objective	Reducing energy consumption and delay	Resource utilization and response time	Reducing energy consumption
Technique	Load balance	Load balance	Energy-resource management and load balance
Reinforcement Learning	Yes	No	No
Optimization Model	Yes	Yes	Yes

- Number of dead electric vehicles (NDEV): it refers to the number of electric vehicles with residual energy levels equal to 0 at the end of simulation time.
- Energy consumption (EC): it refers to the amount of consumed energy by electric vehicles through only the processing operations of the tasks

Algorithm 1: Selecting the suitable vehicle or fog node to execute a task *t* of a fuel vehicle *f*.

```
      ## Fuel vehicle checks the task size.
1.  If size_t ≤ λ then
2.      Q_f ← t
3.      f executes t
4.    Else
      ## f checks its queue
5.    If Q_f is empty then
6.        Q_f ← t
7.        f executes t
8.      Else
          ## f checks the received notifications
9.      If (Noti_1 == 1) ∨ (Noti_2 == 1) then
10.           Q_f ← t
11.           f executes t
12.       Else
13.           f sends a request to the cluster head i.
            ## The cluster head i selects the suitable fog node depending on the load.
14.         If R_i > 1 then
15.             F = min{L_1, L_2, …, L_i}
16.             CH sends a response to f
17.             f forwards t to F through CH or gateway node
18.           Else (i.e. R_i == 1)
19.             CH sends a response to f
20.             f sends t to the available fog node through CH or gateway node
21.         End If
22.     End If
23.   End If
24. End If
```

Algorithm 2: Selecting the suitable vehicle or fog node to execute a task of an electric vehicle.

```
    ## Electric vehicle checks the received notifications.
1.  If Noti_2 == 1 then
2.      If I_i == 1 then
3.          e executes t
4.        Else W ← t
5.          e waits for p seconds
6.            If e received Noti_3 then
7.                e sends a request to CH
8.                Go to Step 17
9.              Else
10.               e executes t
11.           End If
12.     End If
13.   Else
14.     e sends a request to CH
15.     Go to Step 17
16. End If
    ## Cluster head checks the task size and selects best fuel vehicle or fog node to execute this task.
17. If (size_t ≤ λ ∧ V_i > 0) ∨ (size_t > λ ∧ R_i == 0) then
18.       If V_i == 1 then
19.           CH sends a response to e
20.           e forwards t to the selected fuel vehicle
21.         Else (i.e. V_i > 1) then
22.           D = min{ ϒ_1, ϒ_2, …, ϒ_i}
23.           CH sends a response to e
24.           e forwards t to D
25.       End If
26.   Else
27.       If R_i == 1 then
28.           CH sends a response to e
29.           e forwards t to the selected fog node through CH or gateway node
30.         Else (i.e. R_i > 1) then
31.           F = min{L_1, L_2, …, L_i}
32.           CH sends a response to e
33.           e forwards t to F through CH or gateway node
34.       End If
35. End If
```

- Percentage of executed tasks (PET): it refers to the tasks that are executed by the fog nodes and fuel and electric vehicles. It is computed by dividing the number of tasks executed completely by the total number of launched tasks and multiplying by 100.

To do the simulation, we used the simulator OMNeT++ 5.0 and SUMO 0.19.0. To make the simulation environment more realistic, the website www.openstreetmap.org is used to import the Baghdad map and add it to SUMO. Table 6 produces completed information about the simulation environment. However, to study the effect of a parameter on the performance of ERTH, PLIFS, and RMOIE, we change its value and fix the values of other parameters as follows:

7.1. Number of fuel vehicles

This scenario studies the effect of the number of fuel vehicles on the performance of ERTH, PLIFS, and RMOIE in executing IoHV tasks and how can help save the energy of electric vehicles. In this scenario, we assume that 500 electric vehicles are moving at speeds arranged from 5 to 10 m/s. This number is fixed in this scenario. Moreover, we used various numbers of fuel vehicles 1200, 1400, 1600, 1800, and 2000 vehicles move at speeds ranging from 10 to 20 m/s to show how they can affect the executing of the tasks. The number of fog nodes is 100 nodes with different computing and storage capabilities spread beside the roads. The numbers launched tasks from the electric vehicles and fuel vehicles are 100 and 200 respectively with different volumes ranging from 50 to 300 MB.

Table 6. Simulation parameters.

Parameters	Details
Processor	13th Gen Intel(R) Core(TM) i7-1355U, 1.70 GHz
Installed RAM	16.0 GB
Operation system	Windows 11 Pro
Simulation software	Omnet++ 5.0 and SUMO 0.19.0.
Mechanisms	ERTH, RMOIE, and PLIFS
Nodes	Fuel and electric vehicles and fog nodes (i.e. RSU)
Connection protocol	IEEE 802.11p
Energy threshold	15 %
Residual energy of electric vehicles	Ranges from 20 to 90 %
Receiving Energy	Random.
Transmission Energy	Random.
Transmission Power	7.5 db
Data payload	256–1024 Byte/packet
Time of simulation	600 s
Simulation iterations	10
Number and speed of vehicles	As shown in scenarios 1 and 2.
Number of tasks	As shown in scenarios 1 and 2.
Task volume	Ranging from 50 to 300 MB
Task importance	Random (high and low)
Number of fog nodes	100
Processor frequency of fog node	5×10^9 cycles/second
Memory of fog nodes	Ranges from 1 to 4 GB.

Fig. 6 shows the number of dead electric vehicles during this scenario by applying ERTH, PLIFS, and RMOIE. Using PLIFS, the vehicles (whether fuel or electric) send periodic information to the SDN controller.

The sending operations of information consume the vehicles' energy. RMOIE does not focus on the vehicle's energy and focuses on the energy consumption in edge computing nodes. Thus, we see that the numbers of dead electric vehicles using RMOIE are equivalent when increasing the number of fuel vehicles. ERTH gives high priority to the fuel vehicles and fog nodes as shown in Algorithm 1 and Algorithm 2 to execute the tasks while it uses the electric vehicle in some cases when there are no available fuel vehicles or fog nodes. Thus, the number of electric vehicles that participate in processing the tasks is low. Also, the proposed clustering technique is efficient because it gives high priority to the fuel vehicles to work as cluster heads. Moreover, it prevents electric vehicles with a minimum value of *energy_threshold* from working as a cluster head. Thus, the number of dead electric vehicles when applying ERTH is low.

Fig. 7 illustrates the total energy consumption of electric vehicles that used to work as cluster heads and gateways, execute tasks, and forward information and tasks using ERTH, PLIFS, and RMOIE. As it is shown, using ERTH, the energy consumption reduces with an increase the fuel vehicles because when there is a large number of fuel vehicles, the probability of running the tasks by electric vehicles reduces. The proposed system uses Algorithm 2 to

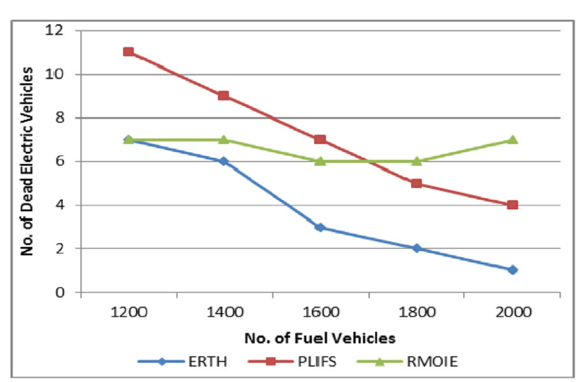

Fig. 6. Number of dead electric vehicles.

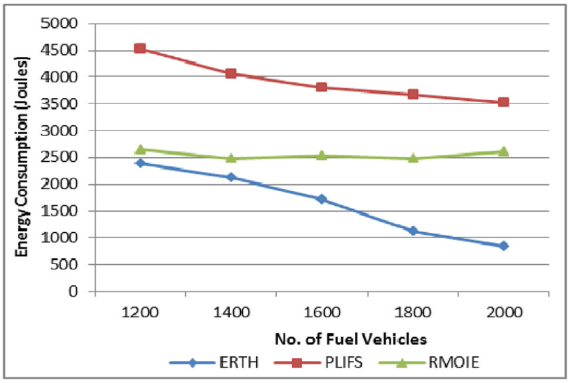

Fig. 7. Energy consumption of electric vehicles.

help electric vehicles in execute tasks by providing resources from fuel vehicles and fog nodes. In addition, the proposed clustering mechanism uses equation (1) to select the fuel vehicle with a maximum number of connected fog nodes and moves at minimum speed to do the roles of cluster head. Moreover, it prevents electric vehicles from playing the role of cluster head except in some rare cases when there is no fuel vehicle. Therefore, this can help save the energy of electric vehicles. In contrast to ERTH, PLIFS and RMOIE do not care about the energy of electric vehicles. In RMOIE, the energy consumption is the same when the number of fuel vehicles increases because it focuses on the energy of fog nodes only. In PLIFS, increasing the number of fuel vehicles increases the number of parked vehicles which means that the probability of running the tasks by these parked fuel vehicles increases. As a result, it saves the energy of electric vehicles. Thus, the energy consumption of electric vehicles using PLIFS reduces with increasing the number of fuel vehicles. However, ERTH is better than PLIFS and RMOIE.

Fig. 8 shows the percentage of executed tasks using ERTH, PLIFS, and RMOIE with increasing the number of fuel vehicles. Increasing the number of fuel vehicles increases the available resources and the probability of executing the tasks. However, RMOIE uses only edge nodes to execute the tasks and as a result, the percentage of executed tasks is not affected when increasing the number of fuel vehicles. The edge nodes may be full sometimes and many tasks will not process. PLIFS exploits the computation resources of fog nodes and parked vehicles only. Therefore, its results in this metric are affected but not significantly. On the other hand, ERTH focuses on exploiting all the available resources in the fuel vehicles and fog nodes to execute the generated tasks by fuel and electric vehicles. Thus, each cluster head determines the closest fog node(s) using equations (11)—(13). Therefore, we see that the percentage of the executed tasks by ERTH is better than that of PLIFS and RMOIE with large numbers of fuel vehicles.

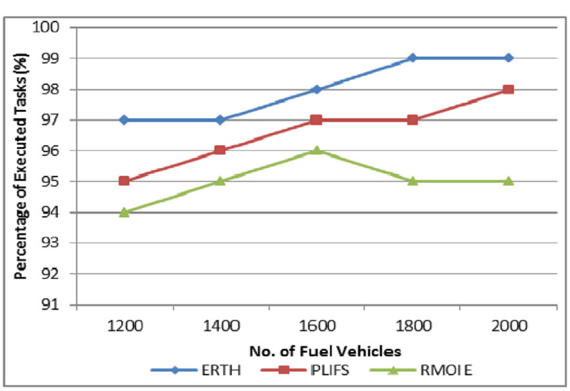

Fig. 8. Percentage of executed tasks.

The performance of resource management techniques is affected by the number of fuel vehicles. When the number of fuel vehicles is low or they are spread in places far from electric vehicles, the electric vehicles will work as cluster heads and execute all the launched tasks. Thus, the energy consumption and number of dead electric vehicles will be increased and the number of executed tasks will be low. However, ERTH produces good results compared to PLIFS and RMOIE because the electric vehicles will execute the small tasks and send the large tasks to the fog nodes. Moreover, electric vehicles execute tasks with high importance levels and add tasks with low degrees of importance to the waiting queue. The tasks stay in this queue for some time p and wait for changes in the cluster. If a fuel vehicle(s) enters the cluster, the electric vehicles send the tasks to that fuel vehicle(s). Otherwise, the electric vehicle is forced to execute those tasks. Thus, ERTH is better than PLIFS and RMOIE.

7.2. Number of tasks

This scenario explains how ERTH, PLIFS, and RMOIE handle the increase in the generated tasks. To do that, numbers of tasks 150, 300, 450, 600, and 750 tasks with different volumes ranging from 50 to 300 MB generated from electric and fuel vehicles. In this scenario, the numbers of fuel and electric vehicles are 500 and 1500 respectively and there are 100 fog nodes with heterogeneous capabilities of computing and storage. The mobility speeds of electric and fuel vehicles are 5—10 m/s and 10—20 m/s respectively.

Fig. 9 illustrates the effect of increasing the number of IoHV tasks on the lifetime of electric vehicles

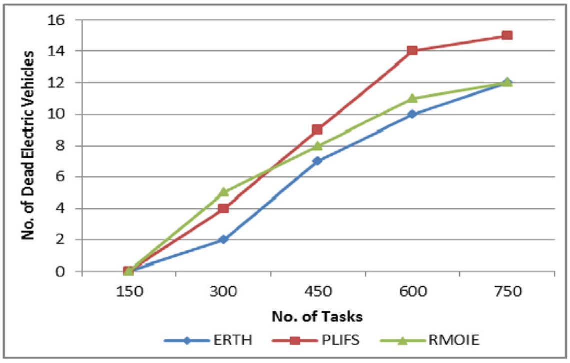

Fig. 9. Number of dead electric vehicles.

when ERTH, PLIFS, and RMOIE are applied. Inefficient exploitation of existing resources will affect negatively the lifetime of the total network. In FLIPS, due to the periodic transfer of information from vehicles to the SDN controller and using some electric parked vehicles to execute some tasks, the number of dead electric vehicles increases. RMOIE does not pay any attention to the vehicle's type and focuses only on how to execute the tasks on the edge nodes with minimum energy and delay. However, increasing the number of tasks increases the demands of finding available edge nodes to execute them which increases the energy consumption and dead electric vehicles. Fortunately, ERTH searches for the available fuel vehicles and sends them the tasks. The proposed system uses equation (1) to provide an optimal number of fog nodes and equations (6)–(8) to balance the load and provide available resources in each cluster. The reason for the increased number of dead electric vehicles using ERTH with an increase in the number of tasks is that electric vehicles are forced to execute some tasks.

Fig. 10 illustrates the total energy consumption of electric vehicles used to execute the tasks using ERTH, PLIFS, and RMOIE, and play the roles of cluster heads and gateways. Using RMOIE, with increasing the number of tasks; the number of repeated demands to find the suitable edge node increases especially when the near ones are full and as a result, the energy consumption maximizes. Using PLIFS, in addition to the frequent exchanging of the vehicles' information, the energy of some electric parked vehicles may be consumed to execute some tasks. Using ERTH, unfair and unexpected distribution of electric and fuel vehicles increases the probability of existing electric vehicles in a geographic area without existing fuel vehicles. Thus, electric vehicles must be used to execute the generated tasks by the vehicles. Moreover, increasing the number of tasks increases the ratio of energy consumption. Fortunately, ERTH and proposed clustering techniques help reduce energy consumption by transferring most tasks to the available fuel vehicles and selecting them to work as cluster heads and gateways. In ERTH, each vehicle uses equation (10) to classify its tasks according to size into small and large. The electric vehicle executes the small tasks if there are no available fuel vehicles and fog nodes and saves the large tasks in the waiting queue for a period of time. If a new fuel vehicle enters into the cluster, the electric vehicle offloads the large tasks to this incoming fuel vehicle. This procedure can help enhance the energy consumption of electric vehicles.

Fig. 11 shows the percentage of executed tasks using ERTH, PLIFS, and RMOIE with the increase in the total number of launched tasks from various IoHV vehicles. With the increase in the number of tasks, the need for a high amount of available resources increases. As a result, several tasks may fail to find the required resources to be processed. In RMOIE, the percentage of executed tasks is small because it uses only the fog nodes which may overload with the large numbers of generated tasks. PLIFS uses only the fog nodes and parked vehicles (if needed) to process tasks which can affect the

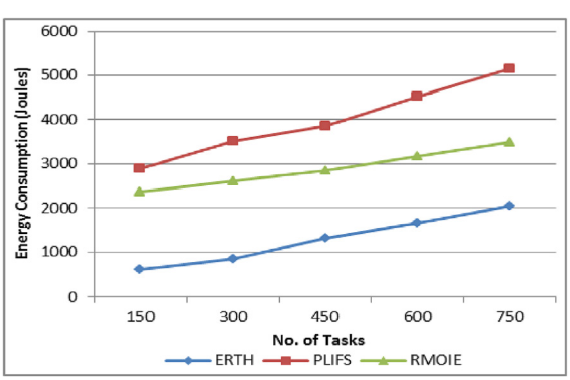

Fig. 10. Energy consumption of electric vehicles.

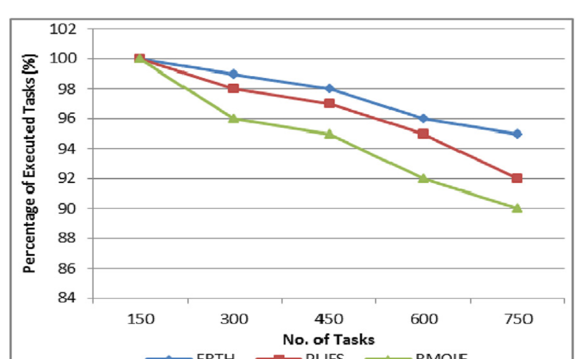

Fig. 11. Percentage of executed tasks.

percentage of executed tasks. ERTH is better than PLIFS and RMOIE in terms of the percentage of executed tasks because it exploits the fog nodes and fuel vehicles and balances the load among the available resources. Moreover, ERTH uses equations (10) and (14) to classify the tasks according to size and importance degree and assigns each one to a suitable resource. As a result, the number of executed tasks is high.

The limitations and drawbacks of the proposed system are:

- Security challenge: the proposed mechanism did not pay any attention to the security challenges that may affect the energy consumption and lifetime of electric vehicles. However, we will focus on this issue in the future.
- Cost: the cost of applying and maintaining the proposed system in environments with no fog nodes is high while it is low in smart cities and advanced environments. However, producing a new optimization approach to determine the optimal number and locations of the fog nodes can produce a good performance and reduce the total cost.
- Implementation: implementing the proposed system in real-world scenarios is difficult due to the lack of the required infrastructure. However, it can be applied in several types of advanced network environments such as smart cities.

8. Conclusions and future works

In IoHV, the vehicles generate various tasks that need to be executed to take a suitable answer. However, the vehicles may fail to execute some tasks due to the lack of computing capabilities. Moreover, electric vehicles are energy-limited. Therefore, executing tasks by these vehicles will affect the electric vehicles' lifetime especially and the network in general. This paper produced an IoHV-fog computing architecture to group the vehicles in clusters and connect them with fog nodes. The clustering of vehicles was done according to our new proposed technique which selected the cluster heads according to vehicle type, connection to fog nodes, energy, and vehicle speed. Moreover, we proposed a new technique to add a new incoming vehicle for a suitable cluster based on the type of incoming vehicle and the number of fuel and electric vehicles. In addition, a novel energy resource management technique for IoHV called ERTH was proposed. It depended on offloading the tasks of electric vehicles to fuel vehicles and fog nodes. It reduced the load on electric vehicles and saved their energy. The simulation showed that ERTH was better than PLIFS and RMOIE. In terms of NDEV, ERTH results were about 47.2 % and 42.4 % enhancement with various numbers of vehicles and 26.19 % and 14.1 % with various mobility speeds compared with PLIFS and RMOIE respectively. EC with ERTH was 58.2 % and 35.8 % with different numbers of fuel vehicles and 61.5 % and 55 % with increasing the mobility speeds less than PLIFS and RMOIE, respectively. PET with ERTH was 2.1 % and 3.09 % with various numbers of fuel vehicles and 1.9 % and 3.18 % with various speeds more than PLIFS and RMOIE, respectively.

The future directions of this research paper are i. Producing a new architecture by merging IoHV with SDN to simplify network management and routing calculation operations. In this architecture, we use a local SDN controller to manage several fog nodes and electric and fuel vehicles in a small geographical area. It collects completed fresh information about these nodes and vehicles and works as a load balancer and resource allocation in that area. The local SDN controllers are connected to a central SDN controller which establishes all connections among the local controllers. It has updated information and balances the load among the geographical area's available resources. ii. Integrating IoHV with the 5G network and network function virtualization technique enhances communications and performance. This type of connection can be established among the SDN controllers and fog nodes to offload the tasks and responses with low response time. The virtualization techniques can be applied in the fog computing layer to increase resource utilization. iii. Producing an approach to classify and schedule the tasks depending on their priorities which is determined according to the task's importance and deadline. iv. Producing a new load balancing technique to reduce the load on the fog nodes by exploiting cloud computing to execute the large volume and delay-tolerance tasks. v. designing a new energy-efficient routing protocol for H-VANET to compute optimal routes to transfer data from an electric or fuel vehicle to another electric or fuel vehicle by exploiting available fuel vehicles as intermediate nodes. vi. focusing on the security risks and potential attacks and producing energy-aware security mechanisms to select the more trusted routes to forward data to the destination.

Ethics information

None.

Funding

The research didn't have a funding.

Acknowledgments

I would like to express my sincere gratitude to my wife lecturer *Rana Ali Shihab* for her invaluable support.

References

[1] D.B. Richardson, Electric vehicles and the electric grid: a review of modeling approaches, impacts, and renewable energy integration, Renew. Sustain. Energy Rev. 19 (2013) 247−254, https://doi.org/10.1016/j.rser.2012.11.042.

[2] Y. Feng, Y. Huang, B. Li, H. Peng, J. Wang, W. Zhou, Connectivity enhancement of E-VANET based on QL-mRSU self-learning energy-saving algorithm, IEEE Access 11 (2023) 3810−3825, https://doi.org/10.1109/ACCESS.2023.3235397.

[3] C. Lv, Y. Liu, X. Hu, H. Guo, D. Cao, F. Wang, Simultaneous observation of hybrid states for cyber-physical systems: a case study of electric vehicle powertrain, IEEE Trans. Cybern. 48 (2018) 2357−2367, https://doi.org/10.1109/TCYB.2017.2738003.

[4] M. Montazeri-Gh, M. Soleymani, Investigation of the energy regeneration of active suspension system in hybrid electric vehicles, IEEE Trans. Ind. Electron. 57 (2010) 918−925, https://doi.org/10.1109/TIE.2009.2034682.

[5] A.J.K. Al-Shaibany, Stability-delay efficient cluster-based routing protocol for VANET, Karbala Int. J. Mod. Sci. 7 (2021) 189−205, https://doi.org/10.33640/2405-609X.3118.

[6] X. He, Z. Ren, C. Shi, J. Fang, A novel load balancing strategy of software-defined cloud/fog in the internet of vehicles, China Commun. 13 (2016) 140−149, https://doi.org/10.1109/CC.2016.7833468.

[7] P. Habib, M. Farhoudi, S. Kazemian, S. Khorsandi, A. Leon-Garcia, A. Leon-Garcia, Fog computing: a comprehensive architectural survey, IEEE Access 8 (2020) 69105−69133, https://doi.org/10.1109/ACCESS.2020.2983253.

[8] J. Santiago, H. Bernhoff, S. Eriksson, S. Ferhatovic, R. Waters, M. Leijon, Electrical motor drivelines in commercial all-electric vehicles: a review, IEEE Trans. Veh. Technol. 61 (2012) 475−484, https://doi.org/10.1109/TVT.2011.2177873.

[9] M.H. Badole, A.D. Thakare, Cluster-based multicast optimized routing in VANETs using elite knowledge-based genetic algorithm, Knowl. Base. Syst. 294 (2024) 154−170, https://doi.org/10.1016/j.knosys.2024.111773.

[10] E. Qafzezi, K. Bylykbashi, E. Spaho, L. Barolli, A new fuzzy-based resource management system for SDN-VANETs, Int. J. Mobile Comput. Multimed. Commun. 10 (2019) 1−12, https://doi.org/10.4018/IJMCMC.2019100101.

[11] H. Yang, X. Xie, M. Kadoch, Intelligent resource management based on reinforcement learning for ultra-reliable and low-latency IoV communication networks, IEEE Trans. Veh. Technol. 68 (2019) 4157−4169, https://doi.org/10.1109/TVT.2018.2890686.

[12] G. Li, Y. Zhang, M. Wang, J. Wu, Q. Lin, X. Sheng, Resource management framework based on the stacklberg game in vehicular edge computing, Complexity 1 (2020) 1−11, https://doi.org/10.1155/2020/8936064.

[13] D. Wang, B. Song, F.R. Yu, P. Lin, X. Du, M. Guizani, Resource management for edge intelligence (EI)-assisted IoV using inspired reinforcement learning, IEEE Internet. Things J. 9 (2022) 12588−12600, https://doi.org/10.1109/JIOT.2021.3137984.

[14] A. Zhu, Y. Wen, An efficient resource management optimization scheme for internet of vehicles in edge computing environment, Comput. Intell. Neurosci. 1 (2022) 1−8, https://doi.org/10.1155/2022/3207456.

[15] A.J. Kadhim, J.I. Naser, Proactive load balancing mechanism for fog computing supported by parked vehicles in IoV-SDN, China Commun. 18 (2021) 271−289, https://doi.org/10.23919/JCC.2021.02.019.

[16] X. Hu, X. Tang, Y. Yu, S. Qiu, S. Chen, Joint load balancing and offloading optimization in multiple parked vehicle-assisted edge computing, Wireless Commun. Mobile Comput. 1 (2021) 1−13, https://doi.org/10.1109/JIOT.2018.2876298.

[17] P. Li, W. Xie, Y. Yuan, C. Chen, S. Wan, Deep reinforcement learning for load balancing of edge servers in IoV, Mobile Network. Appl. 27 (2022) 1461−1474, https://doi.org/10.1007/s11036-022-01972-0.

[18] D. Du, L. Jiang, T. Guo, L. Wu, L. Li, X. Zeng, A dynamic load balancing algorithm for IoV CoMP communications, J. Circ. Syst. Comput. 31 (2022) 143−156, https://doi.org/10.1142/S0218126622502206.

[19] S.A. Darade, M. Akkalakshmi, N. Pagar, SDN based load balancing technique in internet of vehicle using integrated whale optimization method, AIP Conf. Proc. 2469 (2022) 31−43, https://doi.org/10.1063/5.0080349.

[20] M. He, Y. Li, X. Wang, Z. Liu, NOMA resource allocation method in IoV based on prioritized DQN-DDPG network, EURASIP J. Appl. Signal Process. 120 (2021) 1−17, https://doi.org/10.1186/s13634-021-00828-1.

[21] N. Kalsoom, I. Ahmad, R. Alroobaea, M. Raza, S. Khalid, Z. Ahmed, I. Ali, Architecture for resource allocation in the internet of vehicles for cooperating driving system, J. Adv. Transport. 2021 (2021) 22−33, https://doi.org/10.1155/2021/6637568.

[22] Y. Zhang, F. Li, B. Li, Computing resource allocation scheme of IoV using deep reinforcement learning in edge computing environment, EURASIP J. Appl. Signal Process. 21 (2021) 1−19, https://doi.org/10.1186/s13634-021-00750-6.

[23] B. Hazarika, K. Singh, S. Biswas, C. Li, DRL-based resource allocation for computation offloading in IoV networks, IEEE Trans. Ind. Inf. 18 (2022) 8027−8038, https://doi.org/10.1109/TII.2022.3168292.

[24] H. Xi, H. Sun, Resource allocation strategy of internet of vehicles using reinforcement learning, J. Inf. Process. Syst. 18 (2022) 443−456, https://doi.org/10.3745/JIPS.02.0175.

[25] Y. Ni, J. He, L. Cai, J. Pan, Y. Bo, Joint roadside unit deployment and service task assignment for internet of vehicles (IoV), IEEE Internet. Things J. 6 (2019) 3271−3283, https://doi.org/10.1109/JIOT.2018.2882436.

[26] Z. Zhou, P. Liu, J. Feng, Y. Zhang, Computation resource allocation and task assignment optimization in vehicular fog computing: a contract-matching, IEEE Trans. Veh. Technol. 68 (2019) 3113−3125, https://doi.org/10.1109/TVT.2019.2894851.

[27] Q. Peng, Y. Jia, L. Liang, Z. Chen, A task assignment scheme for parked-vehicle assisted edge computing in IoV, in: 93rd Vehicular Technology Conference (VTC2021-Spring), IEEE, Finland. 2021, pp. 113−119, https://doi.org/10.1109/VTC2021-Spring51267.2021.9448735.

[28] B.L. Nguyen, D.D. Nguyen, H.X. Nguyen, D.T. Ngo, M. Wagner, Multi-agent task assignment in vehicular edge computing: a regret-matching learning-based approach, IEEE Trans. Emerg. Top. Comput. Intell. 8 (2024) 1527−1539, https://doi.org/10.48550/arXiv.2203.05281.

[29] X. Wu, S. Zhao, H. Deng, Joint task assignment and resource allocation in VFC based on mobility prediction information, Comput. Commun. 205 (2023) 24−34, https://doi.org/10.1016/j.comcom.2023.04.004.

[30] M. Pang, L. Wang, N. Fang, A collaborative scheduling strategy for IoV computing resources considering location privacy protection in mobile edge computing environment, J. Cloud Comput. 9 (2020) 113−125, https://doi.org/10.1186/s13677-020-00201-x.

[31] J. Chen, T. Li, Internet of vehicles resource scheduling based on blockchain and game theory, Math. Probl. Eng. 22 (2022) 21−34, https://doi.org/10.1155/2022/6891618.

[32] S. Hu, G. Li, W. Shi, LARS: a latency-aware and real-time scheduling framework for edge-enabled internet of vehicles, IEEE Trans. Serv. Comput. 16 (2023) 398−411, https://doi.org/10.1109/TSC.2021.3106260.

[33] M. Feng, H. Yao, J. Li, A task scheduling optimization method for vehicles serving as obstacles in mobile edge

computing based IoV systems, Entropy 25 (2023) 1–23, https://doi.org/10.3390/e25010139.
[34] U. Awada, J. Zhang, S. Chen, S. Li, S. Yang, Resource-aware multi-task offloading and dependency-aware scheduling for integrated edge-enabled IoV, J. Syst. Archit. 141 (2023) 130–143, https://doi.org/10.1016/j.sysarc.2023.102923.
[35] Y. Zhai, W. Sun, J. Wu, L. Zhu, J. Shen, X. Du, M. Guizani, An energy aware offloading scheme for interdependent applications in software-defined IoV with fog computing architecture, IEEE Trans. Intell. Transp. Syst. 22 (2021) 3813–3823, https://doi.org/10.1109/TITS.2020.3044177.
[36] X. Chen, Z. Ma, T. Ma, X. Liu, Y. Chen, Energy-efficient resource allocation for NOMA-enabled internet of vehicles, Wirel. Commun. Mob. Comput. 21 (2021) 101–112, https://doi.org/10.1155/2021/7490689.
[37] F. Chen, M. Liwang, Z. Gao, L. Huang, Energy-efficient data collection and resource allocation in age-aware IoV, Commun. Netw. Broadcast Technol. 7 (2022) 1–11, https://doi.org/10.36227/techrxiv.20373987.v1.
[38] D. Han, J. So, Energy-efficient resource allocation based on deep Q-network in V2V communications, Sensors 23 (2023) 145–159, https://doi.org/10.3390/s23031295.
[39] D. Miao, R. Xu, J. Chen, Y. Dai, An energy-aware technique for task allocation in the internet of things using krill herd algorithm, Int. J. Adv. Comput. Sci. Appl. 14 (2023) 669–675, https://doi.org/10.14569/IJACSA.2023.0140378.
[40] M. Vijarania, S. Gupta, A. Agrawal, M.O. Adigun, S.A. Ajagbe, J.B. Awotunde, Energy efficient load-balancing mechanism in Integrated IoT–fog–cloud environment, Electronics 12 (2023) 851–864, https://doi.org/10.3390/electronics12112543.
[41] M.H. Khoobkar, M.D.T. Fooladi, M.H. Rezvani, M.M.G. Sadeghi, Joint optimization of delay and energy in partial offloading using Dual-population replicator dynamics, Expert Syst. Appl. 216 (2023) 1461–1474, https://doi.org/10.1016/j.eswa.2022.119417.
[42] A. Pakmehr, M. Gholipour, E. Zeinali, ETFC: energy-efficient and deadline-aware task scheduling in fog computing, Sustain. Comput. Inform. Syst. 43 (2024) 398–411, https://doi.org/10.1016/j.suscom.2024.100988.