An Adaptive Vehicle Clustering Algorithm Based on Power Minimization in Vehicular Ad-hoc Networks

Haitao Zhao, *Member, IEEE*, Jiawen Tang, Bamidele Adebisi, *Senior Member, IEEE*, Tomoaki Ohtsuki, *Senior Member, IEEE*, Guan Gui, *Senior Member, IEEE*, and Hongbo Zhu

***Abstract*—In this paper, we propose an adaptive vehicle clustering algorithm based on fuzzy C-means algorithm, which aims at minimizing power consumption of the vehicles. Specifically, the proposed algorithm firstly dynamically allocates the computing resources of each virtual machine in the vehicle, according to the popularity of different virtualized network functions. The optimal clustering number to minimize the total energy consumption of vehicles is determined using the fuzzy C-means algorithm and the clustering head is selected based on vehicles moving direction, weighted mobility, and entropy. Simulation results are provided to confirm that the proposed algorithm can decrease the power consumption of vehicles while satisfying the vehicle delay requirement.**

***Index Terms*—Internet of vehicle, fuzzy C-means, edge computing, power consumption, vehicle clustering.**

I. Introduction

Vehicular ad-hoc networks (VANETs) [1] are attracting extensive attention from both academia and industry. The vehicles in VANETs, using existing wireless communication and sensor technologies, can offer various applications [2]–[7], such as collision avoidance, surrounding information collection, and traffic flow control, etc. On-board unit, trusted authority, and roadside uint (RSU) are the main components of the typical structure of VANETs. However, the typical structure of VANETs faces two following challenges: i) Latency issue: Real-time data collection and processing are critical to many applications in VANETs. For example, an autonomous vehicle needs to quickly collect and process the image data collected by the camera to detect and avoid obstacles or calculate the distances between autonomous vehicles through sensors. However, sending data to the cloud for data processing cannot meet strict end-to-end low-latency requirements in VANETs. ii) Privacy issue: In VANETs, the drivers may not be willing to upload their private information (e.g., their travel routes and their personal driving behavior) to the cloud.

To meet these challenges, mobile edge computing (MEC) technology has been introduced into VANETs, which shifts computing resources from the cloud to the edge of the network, as shown in Fig. 1. Thus, the transmission delay is reduced and the private user data is prevented from sending to the cloud [8]. Thanks to the fast development of hardware, such as cental processing unit and graphic processing units, the processing capability of the vehicles become more and more powerful. Recently, many researchers take a further step and do research on vehicle as a server (VaaS), in which a vehicle can provide computing services for drivers and passengers into itself, but also collaborate with other vehicles [9]. However, VANETs still face some challenges, e.g., highly dynamic topology and the requirements of various quality of service (QoS). These challenges make it difficult to transmit data in VANETs [10], [11]. To deal with these challenges, clustering adjacent vehicles into groups shows a great potential on performance improvement. Besides, processing massive tasks inevitably results in huge power consumption. Compared with the cloud-based VANETs, the power consumption is a significant issue for the edge-based VANETs due to the limited resource at the network edge [12]–[15].

Motivated by the aforementioned backgrounds, we propose an adaptive vehicle clustering algorithm, having the goal of minimizing the total power cost of vehicles in VANETs. The algorithm firstly figures out the neighbor list of each vehicle through angle-based neighbor detection. To efficiently cluster the vehicles, compared with K-means hard clustering, fuzzy C-means (FCM) provides more flexible clustering results, so the fuzzy C-means algorithm is used, and the CH selection is rendered by considering the driving path, entropy, and weighted mobility value of vehiclesVehicle mobility value (used to measure the relative stability of the vehicle and its neighbors). We conduct the simulations to velidate the proposed algorithm. The results of simulations reveal that the proposed algorithm can decrease vehicle power consumption while satisfying the requirement of vehicle delay. The main contributions of this paper are summarized as below.

- We consider a VANET scenario where the vehicles are clustered for efficient data interaction among vehicles. Based on this scenario, we formulate a power

This work was supported by the National Natural Science Foundation of China under grant number 61771252, the National Key Research and Development Program under grant number 2020YFB1806608 and 2019YF-B2103004, the Jiangsu Province Special Fund Project for Transformation of Scientific and Technological Achievements under grant number BA2019058, the Major Natural Science Research Project of Jiangsu Higher Education Institutions under grant number 18KJA510005, the JSPS KAKENHI under Grant JP19H02142, the Jiangsu Provincial Key Research and Development Program under Grant BE2020084-5, the Summit of the Six Top Talents Program of Jiangsu under Grant XYDXX-010, the Program for High-Level Entrepreneurial and Innovative Team under Grant CZ002SC19001. (*Corresponding author: Haitao Zhao*)

H. Zhao, J. Tang, G. Gui, and H. Zhu are with College of Communication and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China (E-mail: zhaoht@njupt.edu.cn; tjw12356@126.com; guiguan@njupt.edu.cn; zhuhb@njupt.edu.cn).

B. Adebisi is with the Department of Engineering, Faculty of Science and Engineering, Manchester Metropolitan University, UK (Email: b.adebisi@mmu.ac.uk).

T. Ohtsuki is with the Department of Information and Computer Science, Keio University, Yokohama 223-8521, Japan (e-mail: ohtsuki@ics.keio.ac.jp).

Manuscript received XXX, XX, 2015; revised XXX, XX, 2015.

minimization problem of the clustered vehicles.

- An adaptive and sustainable vehicle clustering algorithm is proposed. To improve the stability of vehicle clusters, we firstly figure out the neighbor list of each vehicle through angle-based neighbor detection. To efficiently cluster the vehicles, the fuzzy C-means (FCM) algorithm is used, and the CH selection is rendered by considering the driving path, entropy, and weighted mobility value of vehicles.
- The proposed algorithm can dynamically adjust the computing resources of the virtual machine in the intelligent vehicle according to the popularity of the task request flow reaching the on-board edge server, instead of the average allocation of CPU, so as to minimize the overall power loss of the vehicle server.

The rest of this paper is organized as follows. In Section II, we survey the current works of power minimization and vehicle clustering. We introduce the system model and formulate the issue under consideration in Section III. We present the proposed algorithm in detail in Section IV. In Section II, we evaluate the efficiency of the proposed algorithm and finally sum up this paper in Section VI.

II. Related Work

There have been many literatures on the optimization algorithms with the goal of reducing power consumption while meeting the requirements of transmission delay. In [16], the authors investigated a delay-guaranteed and energy-efficient load distribution problem in the Internet of Things(IoT) edge cloud storage framework. A delay load distribution algorithm based on the Lyapunov drift-plus-penalty theory was proposed to achieve the optimal power consumption as well as the delay guarantee. To implement resource aware recommendation, authors in [17] developed an edge based communication mechanism. In [18], an optimization problem was created to jointly minimize packet congestion and energy consumption in a balanced way and the optimal results were obtained by the proposed improved heuristic optimization algorithm for krill population minimizing the total cost of MEC. In [19], authors designed an energy-efficient offloading scheme by combining the multi-access qualities of the 5G heterogeneous networks, which optimized offloading and bandwidth resource allocation at the same time to achieve the minimum energy consumption under delay constraints. In [20], to achieve the optimal result of power consumption, a branch and bound method based on reconstruction linearization technique is proposed, and a greedy heuristic algorithm based on Gini coefficient is proposed, which degrades the mixed integer nonlinear programming problem into a convex problem to reduce computational complexity. In [21], a game-theoretic distributed offloading approach was developed to jointly optimize the offload, in which UE can collaborate with each other for minimizing network overhead in terms of energy consumption and latency. Furthermore, an approximate offloading algorithm was developed to give a quick solution to achieve the optimal power consumption.

Meanwhile, vehicle clustering has also been studied during the past few years. In [22], authors designed a cluster association strategy based on real-time speed for vehicles to leave or join a cluster dynamically. Meanwhile, a collaborative scheduling algorithm was also proposed to select the sender vehicle and the corresponding data items for broadcast. In [23], for maximizing the throughput and minimizing the latency in VANET, dynamic clusters were formed through mathematical optimization solutions. In [24], authors proposed a selection algorithm for cluster head (CH) and a cluster switching algorithm for VANETs to fulfill the prerequisites for services sensitive to throughput and delay. In [25], to help VANET build stable clustering, a center-based clustering algorithm was proposed, which reduced the frequency of vehicle state changes in highway scenes. In [26], authors proposed a sociological pattern and route stability based clustering algorithm by modeling the mobility and the movement mode of vehicles through semi-Markov processes. In [27], an improved K-Harmonic means algorithm was proposed, which jointly considered the relative velocity and distance between cluster members and CHs, as well as the available bandwidth of candidate CHs. In [28], with the purpose of robust transmission, a vehicle clustering optimization algorithm based on Moth Flame clustering was proposed where a nature-inspired Moth Flame algorithm was used to optimize the formation of vehicle clusters. With regard to the application of 5G in vehicular communications, authors in [29] proposed a network slicing architecture of a high-speed railway system based on 5G. In [30], the authors proposed a clustering model in the MEC environment. By considering the self-similarity of the request flows, edge servers are clustered in a cluster to work as a single node, so that traffic flows are processed in the same cluster, which significantly reduces the network processing delay and throughput. In [31], authors modeled the power usage of the MEC server and defined the optimum number of clusters to decrease the MEC server power consumption.

Although the power minimization problem in VANETs has been well studied, most existing works focus on reducing the power of the edge servers, not the vehicles. Besides, the existing works ignore the influence of the vehicle clustering on the power consumption.

III. System Model and Problem Formulation

A. VANET

Considering that the RSUs with limited radio coverage are sparsely deployed and the vehicles with enhanced computing capability, it can be considered to use the idle computing resources to assist the MEC servers in the VANET, e.g., VaaS [32]. These vehicular servers are implemented via network function virtualization technique, rather than a custom hardware device. The realization of network services requires the flows to be sequentially processed by a variety of network functions in accordance with a predetermined logical sequence called service function chain (SFC). Each flow is started by a CH vehicle when creating the Virtualized Network Function (VNF) service chain. CH vehicle determines the route in the SFC link based on the required VNF field, and the VNF information is provided by each flow in its data packet. If the

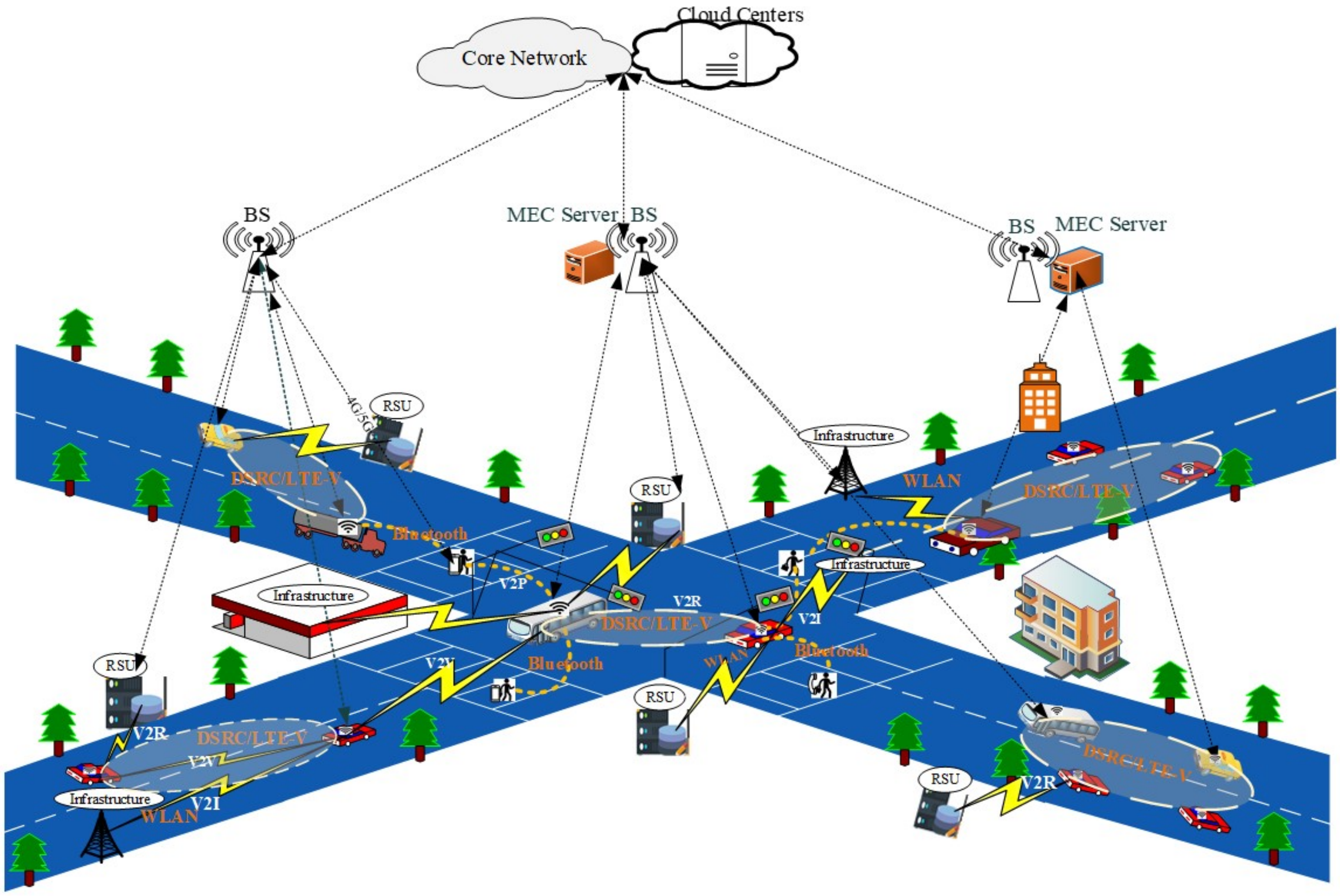

Fig. 1. VANET architecture based on MEC.

cluster does not have the required VNF, it will migrate from another nearest cluster and then the member vehicle creates a VNF chain.

In the considered VANET, the VNFs connects the mobile edge hosts and applications within the data plane through the SFC [31], [33]. Each vehicle (server) cannot contain all kinds of VNFs, that is, the computing power of each VM is different. Therefore, the capacity of the vehicles can be elastically expanded and reduced by dynamically adjusting VNFs. In the considered system model, since the mobile edge application is the injection destination of the inflow, it is assumed that the mobile edge host is set up in each vehicle. Network Functions Virtualization (NFV) service chain includes VNFS chain and mobile edge application [32].

B. Power Consumption Model in VANET

As shown in Fig. 1, R smart vehicles are separate into L non-overlapping clusters. In each cluster, one vehicle is chosen as the CH and the remaining $(R/L-1)$ vehicles are the cluster members. It is assumed that each cluster has at least one CH vehicle and one member vehicle, such that $1 \leq L \leq R/2$. The CH vehicle controls the inflow in the cluster and has the information of VNF distribution. Each vehicle possesses C virtual machines (VMs) and each VM corresponds to one VNF unit to maintain the robustness. Each CH vehicle has the most popular C VNFs and CVNFs is used to represent the VNFs installed in CH vehicles. The member vehicles store other VNFs, which are represented as MVNFs. Thus, the set of VNFs $\mathcal{N} = \{1, 2, \cdots, N\}$ can be separated into two sets, i.e., the set of CVNFs $\mathcal{N}^{(\mathrm{CH})} = \{1, 2, \cdots, C\}$ and the set of MVNFs $\mathcal{N}^{(\mathrm{Mem})} = \{C+1, C+2, \cdots, N\}$. The number of idle VMs is $RC-(L-1)C-N$. If $L \geq (R+1-N/C)$, the CVNFs will occupy $(L-1)C$ VMs and $N^U = \max[N-(R-L+1)C,\ 0]$ MVNFs with the = the least popularity are uninstalled.

Assume that the popularity of VNFs observes the rule of Zipf's law [31]. Intuitively, a small number of VNFs are accessed frequently. Among the VNFs installed in the VMs, the popularity of the n-th commonly used VNF is given by

$$q_n = \frac{\Omega}{n^{\alpha}}, \tag{1}$$

where $\Omega = 1/\sum_{n=1}^{N} 1/n^{\alpha} \approx 1/\int_{1}^{N} 1/n^{\alpha} dn = \frac{1-\alpha}{N^{1-\alpha}-1}$ and α is the Zipf parameter to determine content request patterns. During the average delay cycle $T(L)$, it is assumed that F flows come into the vehicles and a flow includes S service requests. According to [34], using the fractional Brownian motion (fBm) to model the request flow, based on which the

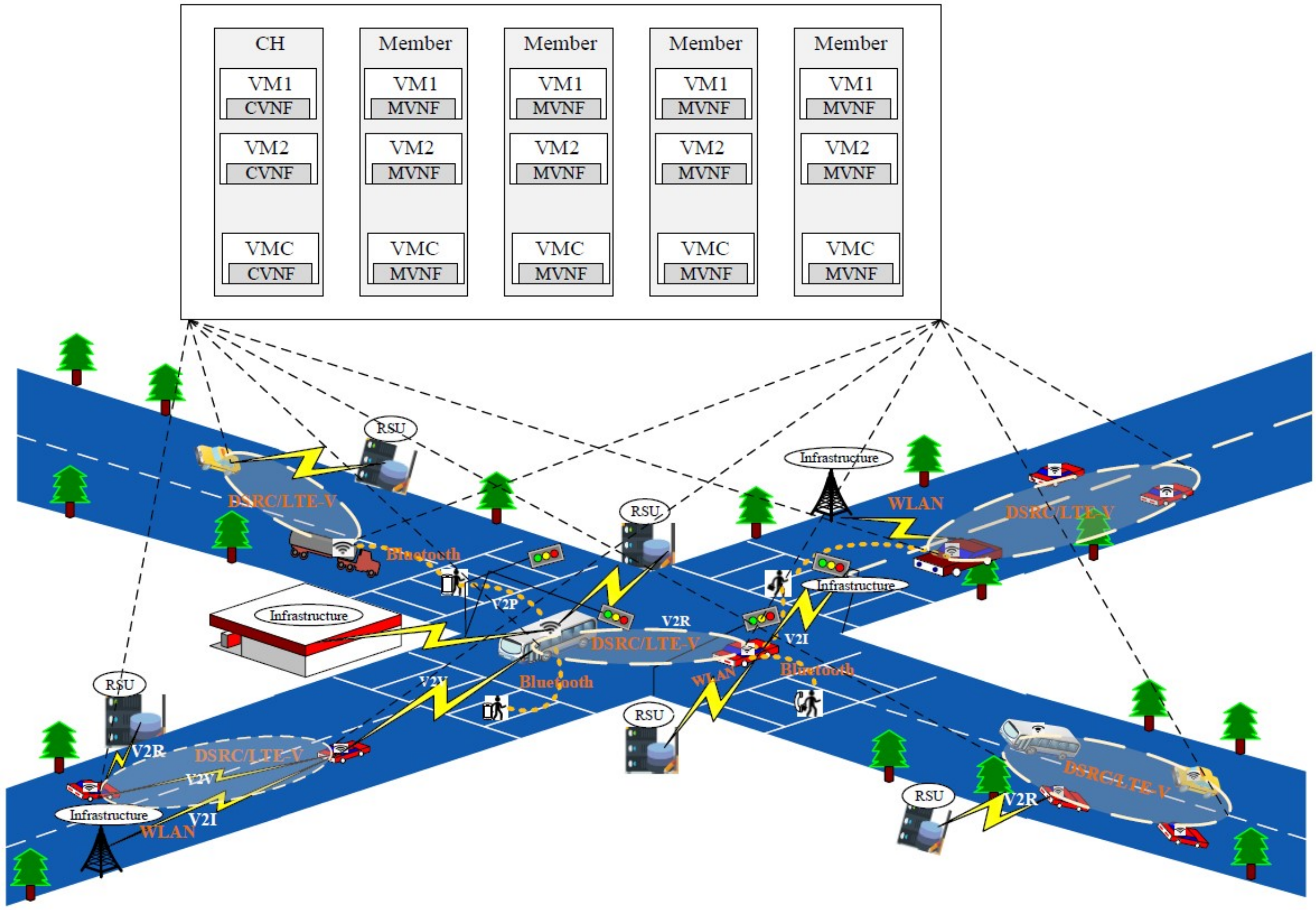

Fig. 2. Vehicle clustering system model based on MEC.

time interval $T(L)$ is given as:

$$T\left(L\right)=2\mu N^{1-\alpha}L^{-1}-2v(RC)^{1-\alpha}L^{\alpha-1}+2vN^{1-\alpha}, \quad (2)$$

where $T(L)$ represents the average delay of each vehicle (server) in processing task request flow, v, μ, and w denote respectively as

$$v=hS\beta/N^{1-\alpha}-1 \quad (3)$$

$$\mu=\frac{hSw^{-1}F(O^{(\mathrm{VNF})}-\frac{\hat{r}}{\mathcal{H}})^{\frac{2\mathcal{H}}{(\mathcal{H}-2)}}}{N^{1-\alpha}}-1 \quad (4)$$

$$w=(-2\sigma^{2}\ln\rho_{f})^{1/(2\mathcal{H}-2)}/(1-\mathcal{H}) \quad (5)$$

and ρ_f represents the fBm model's overflow probability, h represents the average number of hops between the vehicles, $O^{(\mathrm{VNF})}$ represents the average capacity of the VNF service chain, $\hat{r}$ is the average inflow rate, β indicates the deterministic sum of processing delay, transmission, and propagation, σ represents the mean standard deviation of the request flow, and $\mathcal{H}$ represents the fBm model's Hurst parameter.

C. Power Consumption Model

The power consumption of CH vehicles and member vehicles constitutes the total power consumption of the system, i.e.,

$$P_{\mathrm{v}}(L)=LP\left(l^{(\mathrm{CH})}\right)+(R-L)P\left(l^{(\mathrm{Mem})}\right), \quad (6)$$

where $l^{(\mathrm{CH})}$ and $l^{(\mathrm{Mem})}$ are the average CPU loads of a CH vehicle and a member vehicle per second, respectively. According to the dynamic voltage frequency system model of CPU load [35], the power consumption $P\left(u\right)$ for processing depends on the CPU load u, i.e.,

$$P(u)=p_{i}+\frac{P_{b}-P_{i}}{2}\left(1+u^{3}-\exp\left(-\frac{u^{3}}{a}\right)\right). \quad (7)$$

where P_b and P_i are the power consumption of a vehicle when the CPU is fully utilized and idle, and a is level of utilization of power consumption model. Thus, the average CPU load of the vehicles needs to be estimated before calculating the power consumption. According to [31], we can measure the average CPU load of a vehicle using the average number of requests sent to a vehicle during one average delay cycle.

1) Average CPU load of CH vehicles: Given that the average number of requests to each CVNF n of the CH vehicle during $T(L)$ is ${r_n}^{\mathrm{CH}}=\frac{FS}{L}q_n$, the probability of the n-th CVNF is busy per second is computed:

$${p_n}^{(\mathrm{CH})}=\frac{{r_n}^{(\mathrm{CH})}t_n}{T\left(L\right)}=\frac{F\eta_n q_n}{L}, \quad (8)$$

where $t_n=\frac{\eta_n T(L)}{S}$ represents the processing time when one request reaches the n-th CVNF and η_n represents the part of time used to handle a request in CVNF n. Therefore, the CPU load of each CVNF per second is ${l_n}^{(\mathrm{CH})} = p_n^{(\mathrm{CH})}\widehat{l}_n^{(\mathrm{CH})}$, where $\widehat{l}_n^{(\mathrm{CH})}$ is the CPU load generated when CVNF n is working. Without losing versatility, we assume that the CPU runs at full power, that is, $\widehat{l}_n^{(\mathrm{CH})} = 1$. Hence, the CPU load generated by the CH vehicle is expressed as:

$$l^{(\mathrm{CH})} = \sum_{n=1}^{C} {l_n}^{(\mathrm{CH})}. \tag{9}$$

2) Average CPU load of member vehicles: In addition to the operation of the MVNFs, the replacement process of the MVNFs also causes the CPU load. Specifically, when the required VNF is not available, the MVNF with the lowest popularity in the cluster will be replaced with the one required by the flow. This dynamic replacement process will occupy the CPU resources of the member vehicles, and additional power consumption generates. Thus, the average CPU load of member vehicles is measured across different processes, including average CPU load of replacing MVNFs $l^{(\mathrm{Rep})}$, and average CPU load of operating the MVNFs $l^{(\mathrm{Ope})}$, i.e., $l^{(\mathrm{Mem})}{=}l^{(\mathrm{Ope})} + l^{(\mathrm{Rep})}$.

Define $Q^{(\mathrm{U})} = \sum_{n\in N^{(\mathrm{U})}} q_n$ as the total popularity of the $N^{(\mathrm{U})}$ uninstalled MVNFs [31]. The average number of requests to MVNFs of the member vehicles during $T(L)$ is $R^{(U)} = FSQ^{(\mathrm{U})}$. Therefore, the average CPU load required for the replacement of the MVNFs is given by:

$$l^{(\mathrm{Rep})} = \frac{\varphi FSQ^{(U)}\hat{l}^{(\mathrm{Req})}}{R-L}, \tag{10}$$

where φ represents the proportion of MVNF replacement time during $T(L)$, $\hat{l}^{(\mathrm{Req})}$ is the CPU load produced when replacing an MVNF by member vehicles. The sum of the popularity of all the MVNFs that have been already installed in member vehicles is $1 - Q^{(\mathrm{CH})} - Q^{(\mathrm{U})}$ with $Q^{(\mathrm{CH})} = \sum_{n=1}^{C} q_n$, then the total popularity of each member vehicle is:

$$P^{(\mathrm{Mem})} = \frac{1 - Q^{(\mathrm{CH})} - Q^{(\mathrm{U})}}{R-L}. \tag{11}$$

Given the processing time $t_{n_{ic}} = \eta_{n_{ic}} T(L)/S$, where n_{ic} is the MVNF i in cluster c. Let $p_{n_{ic}}$ be the popularity of MVNF n_{ic}, then the probability of the MVNF n_{ic} busy per second is expressed by:

$$p_{n_{ic}}^{(\mathrm{MVNF})} = \frac{FSp_{n_{ic}}}{T(L)} \cdot t_{n_{ic}} = F\eta_{n_{ic}} p_{n_{ic}}, \tag{12}$$

where $i \in \{1, 2, \cdots, R-L\}$ and $c \in \{1, 2, \cdots, C\}$.

Define $\hat{l}_n^{(\mathrm{Mem})}$ as the load of the CPU that a request to join MVNF produces. Similar to $\hat{l}_n^{(\mathrm{CH})}$, $\hat{l}_n^{(\mathrm{Mem})}$ is also assumed to be 1. The average CPU load of the single MVNF per second is $l_{n_{ic}} = p_{n_{ic}}^{(\mathrm{MVNF})}\hat{l}_n^{(\mathrm{Mem})} = Fp_{n_{ic}}\eta_{n_{ic}}$. Finally, the average CPU load of the MVNFs initially installed in member vehicles in the c-th cluster could be calculated by $l_n^{(\mathrm{Ope})} = \sum_{c\in\{1,2,\cdots,C\}} l_{n_{ic}} = \sum_{c\in\{1,2,\cdots,C\}} F\eta_{n_{ic}} p_{n_{ic}}$ because of the non-interfering among the MVNFs running in the member vehicles. Therefore, when a member vehicle is operating, its average CPU load per second is:

$$l^{(\mathrm{Ope})} = \sum_{i\in\{1,2,\cdots,R-L\}} l_i^{(\mathrm{Ope})} = \sum_{c\in\{1,2,\cdots,C\}} F\eta_{n_{ic}} P^{(\mathrm{Mem})}. \tag{13}$$

From the above analysis, the power consumption calculation of the system is mainly divided into two parts, including the power consumption of CH vehicles $l^{(\mathrm{CH})}$ and the power consumption of member vehicles $l^{(\mathrm{Mem})} = l^{(\mathrm{Ope})} + l^{(\mathrm{Rep})}$. Therefore, the adaptive power consumption optimization model can be transformed into the following convex optimization problem, that is, the considered power minimization problem in this paper is formulated as follows:

$$\begin{aligned} &\min_{L} LP\left(l^{(\mathrm{CH})}\right) + (R-L)P\left(l^{(\mathrm{Ope})} + l^{(\mathrm{Rep})}\right) \\ &\text{s.t.}\quad 0 \le l^{(\mathrm{CH})}, l^{(\mathrm{Rep})} + l^{(\mathrm{Ope})} \le 1 \\ &\qquad\ T(L) \le T^{(\mathrm{Req})},\ 1 \le L \le \tfrac{R}{2} \end{aligned} \tag{14}$$

where $T^{(\mathrm{Req})}$ is the constraint of the average delay. The constraint $1 \le L \le R/2$ is introduced to ensure that there is at least one CH vehicle and one member vehicle in each cluster.

To minimize vehicles power consumption, the CPU load of the CH and member vehicles, $l^{(\mathrm{CH})}$ and $l^{(\mathrm{Mem})}$, should be optimized. According to the above, $l^{(\mathrm{CH})} = \sum_{n=1}^{C} {l_n}^{(\mathrm{CH})} = \sum_{n=1}^{c} F\eta_n q_n/L$, minimize the load of CH vehicle, that is, solve $\min\sum_{n=1}^{C}\eta_n q_n{=}\min\left(\eta_1 q_1 + \eta_2 q_2 + \cdots + \eta_C q_C\right)$, according to the Cauchy inequality theorem, we have

$$\begin{aligned} &\left(q_1\eta_1 + q_2\eta_2 + \cdots + q_C\eta_C\right)\left(\tfrac{1}{\eta_1} + \tfrac{1}{\eta_2} + \cdots + \tfrac{1}{\eta_C}\right) \\ &\ge \left(\sqrt{q_1} + \sqrt{q_2} + \cdots + \sqrt{q_C}\right)^2 \end{aligned} \tag{15}$$

In this paper, we assume $\eta_n = \eta$, $\forall n$. Hence, the minimization problem of the CPU load of the CH vehicles is given as

$$\min l^{(\mathrm{CH})} = \frac{F\eta\left(\sqrt{q_1} + \sqrt{q_2} + \cdots + \sqrt{q_C}\right)^2}{LC}. \tag{16}$$

Similarly, the minimization problem of the CPU load of the member vehicles, produced by the operation of the deployed MVNFs per second, is given as

$$\min l^{(\mathrm{Ope})} = \frac{F\eta\left(\sqrt{q_1} + \sqrt{q_2} + \cdots + \sqrt{q_C}\right)^2}{(R-L)\,C}. \tag{17}$$

Therefore, the total power minimization problem is reformulated as:

$$\begin{aligned} &\min_{L} LP\left(\min l^{(\mathrm{CH})}\right) + (R-L)P\left(\min l^{(\mathrm{Ope})} + \tfrac{\varphi FSQ^{(U)}}{R-L}\right) \\ &\text{s.t.}\quad 0 \le l^{(\mathrm{CH})}, l^{(\mathrm{Rep})} + l^{(\mathrm{Ope})} \le 1 \\ &\qquad\ T(L) \le T^{(\mathrm{Req})},\ 1 \le L \le \tfrac{R}{2}. \end{aligned} \tag{18}$$

IV. The Proposed Vehicle Clustering Algorithms

A. Selection of Optimal Clustering Value L Based on Energy Consumption

K-means algorithm is efficient for the division of large data sets, and the time complexity is nearly linear, but it is sensitive to the initially selected mean vector, has a great impact on the final clustering results, and is easy to fall into the local optimal solution. Although the effect of fuzzy c-means algorithm is better than k-means algorithm in practical application, because it is a variant of K-means algorithm, it still needs to specify the clustering value to cluster the samples in advance, which can not completely reduce the impact of the initial selection point on the clustering results. Therefore, this paper proposes a model based on the optimal energy efficiency to determine the optimal clustering value L of vehicles, which replaces the FCM algorithm to determine the clustering value by traversing all the desirable values, which reduces the complexity of the algorithm and the response time of vehicle clustering.

To determine the optimal cluster value L, it is necessary to ensure that (14) is a convex, so as to effectively calculate the optimal number of clusters and minimize the power consumption of vehicles. Therefore, this optimization problem needs to be solved in two cases. Under the condition of meeting the convexity of the power consumption model, the optimal number of clusters is determined by convex optimization. When the model does not meet the convexity condition, the optimal number of clusters L is found by traversing the search algorithm. The selection process of the optimal clustering value L is shown in **Algorithm 1**.

Algorithm 1 Selection of optimal clustering value L.

Phase 1: Define constraint set Λ
if $\Lambda \subset \emptyset$ **then**,
end if
Phase 2: Determine the optimal number of clusters L
if $1 \leq N \leq (1 + R/2)\, C$ **then**
    $L_{conv} = \arg_L \left[\frac{\partial P_v(L)}{\partial L} = 0 \right]$
    if $P_v(\lfloor L_{conv} \rfloor) > P_v(\lceil L_{conv} \rceil)$ **then** $L_{temp} \leftarrow \lceil L_{conv} \rceil$
    else $L_{temp} \leftarrow \lfloor L_{conv} \rfloor$
    end if
    if $L_{temp} \in \Lambda$ **then** $L_{opt} \leftarrow L_{temp}$
    else if $L_{temp} \leq \min \Lambda$ **then** $L_{opt} \leftarrow \min(\Lambda)$
    else $L_{opt} \leftarrow \max(\Lambda)$
    end if
else find L_{opt} in set Λ by exhaustive search algorithm
end if
return L_{opt}

B. Angle-based Neighbor Detection

Before vehicle clustering, we need to build a list of neighbors for each vehicle, which is based on information collected by using the Global Positioning System (GPS) receiver. The collected information includes accurate speed, exact time, and real-time 3D geographic location (longitude, latitude, and altitude). When performing the vehicle clustering, the vehicles traveling in the opposite directions cannot be grouped into the same cluster because they will lose contact with each other. Thus we group these vehicles according to their moving direction.

It is necessary to judge whether the vehicles are driving in the same direction, which can be determined by calculating the angle between the vehicle's velocity vectors. As shown in Fig. 2, (x_{r_1}, y_{r_1}) and (x_{r_2}, y_{r_2}) respectively represent vehicles r_1 and r_2 locations at time t, whereas $(\hat{x}_{r_1}, \hat{y}_{r_1})$ and $(\hat{x}_{r_2}, \hat{y}_{r_2})$ represent the locations at the time $t + \Delta t$. According to the cosine theorem, the angle θ between two specified velocity vectors is computed as,

$$\theta = \arccos\left(\frac{\Delta x_{r_1} \times \Delta x_{r_2} + \Delta y_{r_1} \times \Delta y_{r_2}}{\sqrt{{\Delta x_{r_1}}^2 + {\Delta y_{r_1}}^2} \times \sqrt{{\Delta x_{r_2}}^2 + {\Delta y_{r_2}}^2}} \right), \tag{19}$$

where $\Delta x_{r_i} = \hat{x}_{r_i} - x_{r_i}$, $\Delta y_{r_i} = \hat{y}_{r_i} - y_{r_i}$, $i = 1, 2$.

In most existing algorithms of vehicle clustering, the vehicles are thought to be on different roads, even when their angle directions are a little different. This assumption is not applicable to real road scenarios. In this paper, we assume that two vehicles with the included angle $\theta \leq \phi$ can be considered to be moving in the same direction, where ϕ is the angle threshold. Here, we set the threshold value $\phi = 18°$ [37].

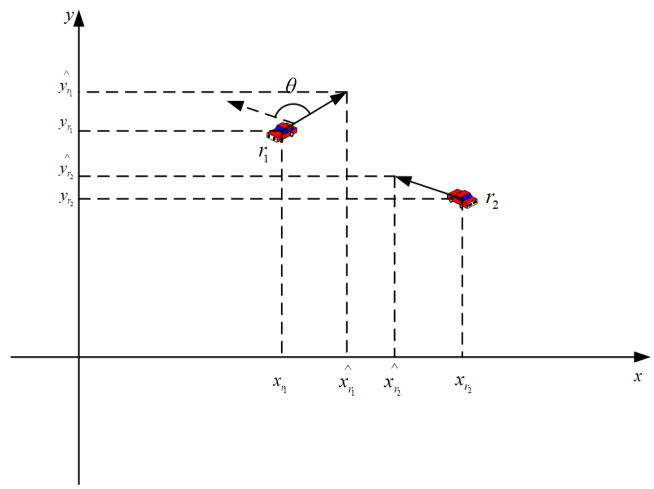

Fig. 3. Calculation of vehicle movement angle.

C. Clustering Process of Vehicles

In a cluster-based VANET, a cluster has one CH and multiple cluster members, and each vehicle can only be allocated to one cluster to ensure that clusters do not intersect each other [38]. In this paper, the FCM algorithm is used for vehicle clustering, which is mainly based on speed, distance, and movement direction. As a variant of K-means algorithm, the FCM algorithm is different from the hard partition of K-means, the membership degree of each sample point in the sample set is obtained by optimizing the objective function, and the fuzzy partition is carried out according to the membership degree.

We define a fuzzy partition matrix $\mathbf{U}$, in which the (r, l)-th element u_{rl} represents the membership of vehicle r to a cluster l. The larger the value of u_{rl} is, the higher the membership degree. In addition, u_{rl} satisfies the following constraints:

$$\begin{aligned} &\sum_{l=1}^{L} u_{rl} = 1, \\ &\sum_{r=1}^{R} u_{rl} > 0, \\ &l = 1, 2, \cdots, L, \\ &r = 1, 2, \cdots, R \end{aligned} \tag{20}$$

Define $\mathbf{M} = \{\mu_1, \mu_2, \cdots, \mu_L\}$ as the mean vector of the clusters, and the Euclidean distance $d_{ij} = \sqrt{(x_j - \mu_i)(x_j - \mu_i)^T}$ is used to represent the paired distance between the vehicle dataset and the mean vector. The objective function is:

$$J=\sum_{r=1}^{R}\sum_{l=1}^{L}{u_{rl}}^{e}d_{rl}^{2}, \tag{21}$$

where e is the fuzzy parameter, generally $e=2$. The Lagrange multiplier method is used and the Lagrange dual function is formulated as

$$\mathcal{L}=\sum_{r=1}^{R}\sum_{l=1}^{L}u_{rl}^{e}d_{rl}^{2}-\lambda\left(\sum_{l=1}^{L}u_{rl}-1\right) \tag{22}$$

Based on $\partial\mathcal{L}/\partial\lambda=0$ and $\partial\mathcal{L}/\partial u_{rl}=0$, we have

$$u_{rl}=\frac{1}{\sum\limits_{c=1}^{L}\left(\frac{d_{rl}}{d_{cl}}\right)^{\frac{2}{e-1}}}, \tag{23}$$

$$\mu_r=\frac{\sum_{l=1}^{R}u_{rl}^{e}x_l}{\sum_{l=1}^{R}u_{rl}^{e}}. \tag{24}$$

Divide the dataset $X=\{x_1,x_2,\cdots,x_R\}$ containing the attribute values of R vehicles into L clusters. Dataset X includes three attributes: vehicle speed, vehicle distance, and angle. The cluster division is characterized by matrix $\mathbf{U}$. When $u_{rl}=0$, it means that vehicle r does not belong to cluster l; When $u_{rl}=1$, the representative vehicle r is included into cluster l. The larger the value of u_{rl}, the higher the membership of vehicle r to cluster l. The steps of clustering vehicles are shown in **Algorithm 2**.

Algorithm 2 The process of clustering based FCM.

Input: dataset $X=\{x_1,x_2,\cdots,x_R\}$, number of clusters L;
randomly select L samples from the dataset X as the initial mean vector $\{\mu_1,\mu_2,\cdots,\mu_L\}$;
Initialize the membership matrix with $u_{rl}\in(0,1)$
repeat
  $C_r=\emptyset\,(1\le r\le L)$
  for $l=1,2,\cdots,R$ **do**
    calculate u_{rl} of each data in dataset X for cluster C_r
    calculate the Euclidean distance between x_l and $\mu_r\,(1\le r\le L)$
    calculate the value function (21) & determine x_l belongs to which cluster: $\lambda_l=\arg\min_{r\in\{1,2,\cdots,L\}}J$
    cluster x_l: $C_{\lambda_l}=C_{\lambda_l}\cup\{x_l\}$
  end for
  for $r=1,2,\cdots,L$ **do**
    re-calculate the membership of x_l to cluster C_r
    calculate the new mean vector μ_r' of each cluster
    if $\mu_r'\neq\mu_r$ **then**
      update the mean vector μ_r to μ_r'
    else
      maintain the mean vector μ_r
    end if
  end for
until all the mean vectors are refreshed
Output: $C=\{C_1,C_2,\cdots C_L\}$.

D. CH Election

1) Movement direction: After receiving the Hello message and the message from its neighbor r', vehicle r will determine whether its neighbor is close or far away by calculating the distance between the two messages.

$$\vec{P}_{rr'}(t)=\vec{p}_r(t)-\vec{p}_{r'}(t). \tag{25}$$

where $\vec{p}_r(t)$ represents the position of vehicle r at time t. Note that $|\vec{P}_{rr'}(t)|\ge|\vec{P}_{rr'}(t+1)|$ indicates that the vehicles r and r' get closer to each other; otherwise be farther.

2) Information entropy: According to [38], the measurement value of vehicle stability can be obtained by calculating the entropy. Variable α is the feature of the variable, and its expression is as follows:

$$\alpha_{r,r'}=\frac{1}{D}\sum_{i=1}^{D}\left|\overrightarrow{p}_{r,r'}(t_i)\right|, \tag{26}$$

where D represents the number of times the vehicle broadcasts messages within the time interval Δt, t_i is the discrete time point of vehicle broadcast message. The formula for calculating the entropy of vehicle r is given by

$$H_r(t,\Delta t)=-\frac{\sum_{l\in F_r}Q_l(t,\Delta t)\log_2 Q_l(t,\Delta t)}{\log_2 C(F_r)}, \tag{27}$$

where F_r represents the collection of vehicle r and its neighboring vehicles r'; $C(F_r)$ is the number of vehicles to the vehicle r; for the variable feature α of all vehicles, we take its normalized value, namely

$$Q_l(t,\Delta t)=\frac{\alpha_{r',r}}{\sum_{i\in F_r}\alpha_{r',r}}. \tag{28}$$

3) Vehicle mobility values: According to [39], the mobility value of vehicles can be defined to reflect the number of vehicles traveling around a vehicle inside its scope of transmission. At time t, define the mobility value of vehicle r_i:

$$S_{r_i}(t)=\frac{In_{r_i}(t)+Out_{r_i}(t)}{N_{r_i}(t-1)}, \tag{29}$$

where $In_{r_i}(t)$ and $Out_{r_i}(t)$ are used to represent the number of vehicles entering and leaving the list of neighbors during the time interval from $t-1$ to t, respectively, and $N_{r_i}(t-1)$ represents the number of vehicles that belong to the vehicle r_i's neighbor list at the previous broadcast time $t-1$.

4) Vehicle weighted mobility value: The vehicle r_i's weighted mobility value at time t is

$$S_{r_i}^{w}(t)=w\cdot S_{r_i}(t)+(1-w)\cdot S_{r_i}^{w}(t-1). \tag{30}$$

According to the weighted clustering algorithm [40], the weighted sum of all vehicles in the cluster is calculated. The vehicle with the minimum weighted sum is the cluster head. Hence, the weighted sum of vehicles is calculated as:

$$W_r(t)=-w_1\cdot\left|\overrightarrow{P_{r,r'}}\right|+w_2\cdot S_{r,r'}^{w}(t)-w_3\cdot H_r(t,\Delta t), \tag{31}$$

where w_i represents the weight factors, with $\sum_{i=1}^{3}w_i=1, w_i>0$.

V. Simulation Results And Discussion

This paper uses MATLAB for simulation to compare the power consumption under various cluster numbers to test the output of the proposed algorithm. The performance is tested on a random network composed of 100 vehicles, which are evenly and randomly distributed on a standardized circular area with F flows. Table I lists the parameter values of simulation.

TABLE I
PARAMETER VALUES

Symbol	Values	Specification
S	30	the average number of requests in each flow
C	5	the number of VMs
a	0.3	level of utilization of power consumption model
P_b	274 W	the power consumption of a vehicle when the CPU is fully utilized
P_i	62.6 W	the power consumption of a vehicles when the CPU is idle
φ	0.01	the proportion of time to replace MVNF during $T(L)$
$\hat{l}^{(\mathrm{Rep})}$	0.005	CPU load generated by member vehicles when replacing an MVNF
α	0.8	parameter of Zipfs law
$l^{(\mathrm{Cap})}$	10 Gbps	the average capacity of the VNF service chain
β	30 ms	the deterministic sum of propagation
r	1-3 Gbps	the average inflow rate
h	$64/45\pi$	the average number of hops counts between the vehicles
H	0.8	the Hurst parameter of the fBm model
η	0.0006	the portion of time dedicated to process a request in the VNF

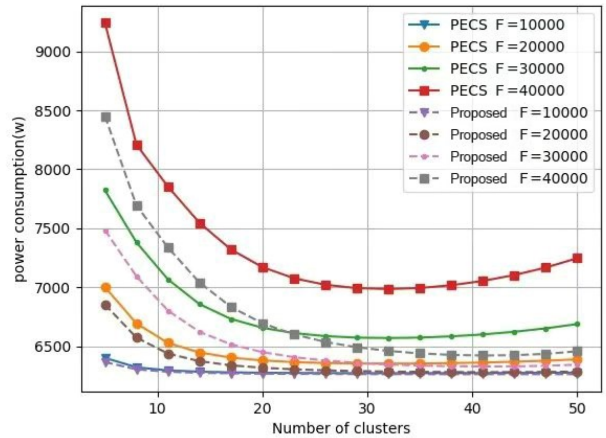

Fig. 4. Power consumption of PECS and the proposed algorithm with different number of clusters when $N = 400$.

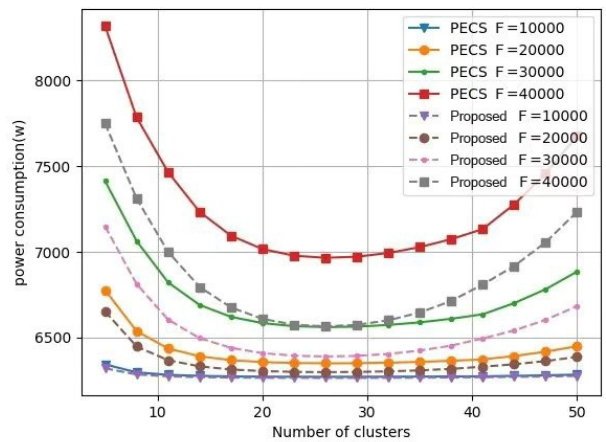

Fig. 5. Power consumption of PECS and the proposed algorithm with different number of clusters when $N = 600$.

The power consumption of the power efficient clustering scheme (PECS) [31] algorithm and the proposed algorithm over different numbers of clusters is shown in Fig. 4. It is visible from Fig. 4 that, as the number of clusters increases at the outset, the power consumption of both the PECS algorithm and the proposed algorithm decreases. This is because the increase in the number of CH vehicles makes it easier to process the requests with high popularity. However, when the number of clusters increases to a threshold, the power consumption of both the PECS and the proposed algorithms increases. This observation suggests that the requests with small popularity cannot be processed efficiently. Besides, we find that the proposed algorithm achieves lower power consumption than the PECS algorithm no matter what number of clusters are chosen. It is also observed that the greater the value of F is, the more power consumption can be reduced by the proposed algorithm. This is because, in the PECS algorithm, the VM allocates CPU resources in the way of average allocation. When the conventional task request flows reach the edge server, the low popularity VNF still occupies the CPU resources of the VM, and the idle CPU resources cannot be dynamically allocated to a large number of high popularity task flows. The algorithm proposed in this paper is based on the different popularity of VNF dynamically adaptive allocate CPU resources of VM.

In Fig. 5, the number of VNFs N is set to 600. As the number of clusters increases, the power consumption of both the PECS and proposed algorithms first decreases and then increases, which shows a similar trend to Fig. 4. However, when the value of L increases to a threshold, the power consumption of both the PECS and proposed algorithms increases much more than that in Fig. 4 with $N = 400$. It means that the number of types of the requests is bigger than that of the VNFs contained in the vehicles. Hence, additional power consumption is required for the VNF replacement.

In Figs. 6 and 7, we compare the clustered NFV service chaining (cNSC) [30] algorithm, the PECS algorithm, and the proposed algorithm in terms of the total power consumption and average delay. As shown in Fig. 6, with the rise in flows, no matter which algorithm is applied, the total power consumption of the vehicles increases. We note that the proposed algorithm's average power consumption is smaller than that of the cNSC and PECS algorithms. Importantly, the larger the number of task request flows is, the more obvious the performance gap between the proposed algorithm and the two baseline algorithms. In addition, when $F > 30,000$, the power consumption of the vehicles increases rapidly, consistent with the theoretical explanation that the CPU load is too high due to the large number of task requests.

In Fig. 7, we can see that no matter if the number of flows varies, the average delay $T(L)$ of the PECS and proposed algorithms can be kept at an appropriate amount, $T^{(\mathrm{Req})}$. When the number of flows is small, the average delay of the PECS and proposed algorithms is higher than that of the cNSC algorithm. The reason behind this observation is that the PECS and proposed algorithms aim to find the optimal number of

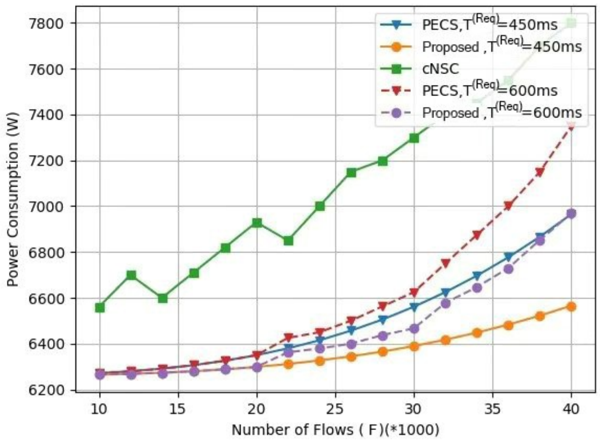

Fig. 6. Power consumption comparison with different number of flows when $N = 400$.

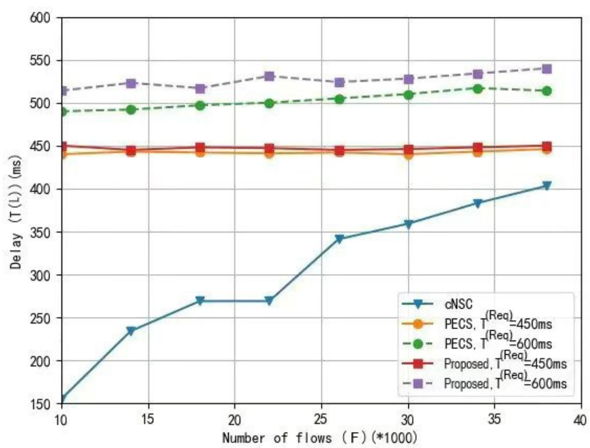

Fig. 7. Average delay comparison with different number of flows when $N = 400$.

clusters to minimize the power consumption. In other words, the cost of reducing the overall power consumption through PECS and proposed algorithms is the increase in the delay of processing tasks, however the cost is within an acceptable range, that is, the delay of task processing is controlled within $T^{(\text{Req})}$.

VI. Conclusion

To decrease the power consumption and improve the duration of vehicle clusters, we propose an adaptive vehicle clustering algorithm based on power minimization in the VANET. The proposed algorithm firstly allocates the computing resource according to the popularity of different VNFs. The optimal clustering number is determined using the fuzzy C-means algorithm and the clustering head is selected based on vehicles moving direction, entropy, and weighted mobility. To validate the performance of the proposed algorithm, we conduct the simulations. The proposed algorithm can decrease vehicles' power consumption while satisfying the requirement of vehicle delay. In future work, we try to introduce the distributed learning methods, such as federated learning approach, into the VANET to realize intelligent and low-complexity vehicle clustering.

References

[1] S. Liu, J. Tang, Z. Zhang, and J. Gaudiot, "Computer architectures for autonomous driving," *Computer*, vol. 50, no. 8, pp. 18–25, 2017.

[2] Y. Kawamoto, T. Mitsuhashi, N. Kato, "UAV-aided information diffusion for vehicle-to-vehicle (V2V) in disaster scenarios," *IEEE Transactions on Emerging Topics in Computing*, doi: 10.1109/TETC.2021.3120551

[3] Y. Zhao, H. Jiang, *et al.*, "Preserving minority structures in graph sampling," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 2, pp. 1698–1708, 2021.

[4] Y. Tang, N. Cheng, W. Wu, M. Wang, Y. Dai and X. Shen, "Delay-minimization routing for heterogeneous VANETs With machine learning based mobility prediction," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3967–3979, Apr. 2019.

[5] F. Tang, B. Mao, *et al.*, "Comprehensive survey on machine learning in vehicular network: technology, applications and challenges," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 2027–2057, thirdquarter, 2021.

[6] G. Gui, M. Liu, *et al.*,"6G: Opening new horizons for integration of comfort, security and intelligence," *IEEE Wireless Communications*, vol. 27, no. 5, pp. 126–132, Oct. 2020.

[7] Y. Zhao, J. Shi, *et al.*, "Evaluating effects of background stories on graph perception," *IEEE Transactions on Visualization and Computer Graphics*, early access, doi:10.1109/TVCG.2021.3107297

[8] Y. Mao, C. You, *et al.*, "A survey on mobile edge computing: The communication perspective," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2322–2358, 2017.

[9] S. Abdelhamid, H. S. Hassanein, and G. Takahara, "Vehicle as a resource (VaaR)," *IEEE Network*, vol. 29, no. 1, pp.12–17, 2015.

[10] Y. Yin, M. Liu, *et al.*,"QoS-oriented dynamic power allocation in NOMA-based wireless caching networks," *IEEE Wireless Commuications Letters*, vol. 10, no. 1, pp. 82–86, Jan. 2021.

[11] Y. Yin, M. Liu, *et al.*, "Cross-layer resource allocation for UAV-assisted wireless caching networks with NOMA," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 4, pp. 3428–3438, Apr. 2021.

[12] X. Cheng and B. Huang, "A center-based secure and stable clustering algorithm for VANETs on highways," *Wireless Communications and Mobile Computing*, vol. 2019, article ID: 8415234, Jan. 2019.

[13] L. Sellami and B. Alaya, "SAMNET: Self-adaptative multi-kernel clustering algorithm for urban VANETs," *Vehicular Communications*, vol. 29, article ID: 100332, June 2021.

[14] X. Duan, Y. Liu, and X. Wang, "SDN enabled 5G-VANET: Adaptive vehicle clustering and beamformed transmission for aggregated traffic," *IEEE Communications Magazine*, vol. 55, no. 7, pp. 120–127, July 2017.

[15] T. Wang, X. Cao, and S. Wang, "Self-adaptive clustering and load-bandwidth management for uplink enhancement in heterogeneous vehicular networks," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 120–127, July 2017.

[16] M. Guo, L. Li, and Q. Guan, "Energy-efficient and delay-guaranteed workload allocation in IoT-edge-cloud computing systems," *IEEE Access*, vol. 7, pp. 78685–78697, 2019.

[17] C.X. Mavromoustakis, G. Mastorakis, and J.M. Batalla, "A mobile edge computing model enabling efficient computation offload-aware energy conservation," *IEEE Access*, vol. 7, no. 1, pp. 102295–102303, 2019.

[18] Y. Yang, Y. Ma, W. Xiang, X. Gu, and H. Zhao, "Joint optimization of energy consumption and packet scheduling for mobile edge computing in cyber-physical networks," *IEEE Access*, vol. 6, no. 1, pp. 15576–15586, 2018.

[19] K. Zhang, Y. Mao, S. Leng, Q. Zhao, L. Li, X. Peng, and Y. Zhang, "Energy-efficient offloading for mobile edge computing in 5G heterogeneous Networks," *IEEE Access*, vol. 4, no 1, pp. 5896–5907, 2016.

[20] P. Zhao, H. Tian, C. Qin, and G. Nie, "Energy-saving offloading by jointly allocating radio and computational resources for mobile edge computing," *IEEE Access*, vol. 5, no. 1, pp. 11255–11268, 2017.

[21] B. Wu, J. Zeng, L. Ge, X. Su, and Y. Tang, "Energy-latency aware offloading for hierarchical mobile edge computing," *IEEE Access*, vol. 7, pp. 121982-121997, 2019.

[22] J. Wang, K. Liu, K. Xiao, C. Chen, W. Wu, V.C. Lee, and S.H. Son, "Dynamic clustering and cooperative scheduling for vehicle-to-vehicle communication in bidirectional road scenarios," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 6, pp. 1913–1924, Jun. 2018.

[23] M. Azizian, S. Cherkaoui, and A.S. Hafid, "A distributed cluster based transmission scheduling in VANET," in *IEEE International Conference on Communications (ICC)*, Kuala Lumpur, Malaysia, 23–27 May, 2016, pp. 1–6.

[24] R. Chai, B. Yang, L. Li, X. Sun, and Q. Chen, "Clustering-based data transmission algorithms for VANET," in *International Conference on Wireless Communications and Signal Processing (WCSP)*, Hangzhou, China, October 24–26, 2013, pp. 1–6.

[25] X. Cheng, B. Huang, and W. Cheng, "Stable Clustering for VANETs on Highways," in *ACM/IEEE Symposium on Edge Computing (SEC)*, Seattle, WA, USA, October 25–27, 2018, pp. 399–403.

[26] L. A. Maglaras, D. Katsaros, "Social clustering of vehicles based on Semi-Markov processes," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 1, pp. 318–332, 2016.

[27] R. Chai, X. Ge, and Q. Chen, "Adaptive K-harmonic means clustering algorithm for VANETs," in *International Symposium on Communications and Information Technologies (ISCIT)*, Incheon, Korea, September 24–26, 2014, pp. 233–237.

[28] M. F. Khan, F. Aadil, and M. Maqsood, "Moth flame clustering algorithm for internet of vehicle (MFCA-IoV) ," *IEEE Access*, vol. 7, no. 1, pp. 11613–11629, 2018.

[29] B. Ai, A.F. Molisch, M. Rupp, and Z. Zhong, "5G key technologies for smart railways," *Proceedings of the IEEE*, vol. 108, no. 6, pp. 856–893, Jun. 2020.

[30] Y. Nam, S. Song, and J.M. Chung, "Clustered NFV service chaining optimization in mobile edge clouds," *IEEE Communications Letters*, vol. 21, no. 2, 2017.

[31] J. Ahn, J. Lee, S. Park, and H.S. Park, "Power efficient clustering scheme for 5G mobile edge computing environment," *Mobile Networks and Application*, vol. 24, pp. 643–652, 2019.

[32] J. Zhang, and K.B. Letaief, "Mobile edge intelligence and computing for the internet of vehicles," *Proceedings of the IEEE*, vol. 108, no. 2, 246–261, 2019.

[33] W. Xia, G. Zheng, Y. Zhu, J. Zhang, J. Wang, and A.P. Petropulu, "A deep learning framework for optimization of MISO downlink beamforming," *IEEE Transactions on Communications*, vol. 68, no. 3, pp. 1866–1880, Mar. 2020

[34] A. Rizk and M. Fidler, "Non-asymptotic end-to-end performance bounds for networks with long range dependent fBm cross traffic," *Computer Networks*, vol. 56, no. 1, pp. 127–141, Jan. 2012.

[35] J. Pouwelse, K. Langendoen, and H. Sips, "Energy priority scheduling for variable voltage processors," in *The International Symposium on Low Power Electronics and Design (ISLPED)*, Huntington Beach, CA, USA, August 6–7, 2001, pp. 28–33.

[36] M. Hadded, P. Muhlethaler, A. Laouiti, and L. Saidane, "A novel angle-based clustering algorithm for vehicular ad hoc networks," *Second International Workshop on Vehicular Adhoc Networks for Smart Cities (IWVSC)*, Kuala Lumpur, Malaysia, 14 August, 2016, pp. 1–5.

[37] V. Naumov, T. R. Gross. "Connectivity-aware routing (CAR) in Vehicular ad-hoc networks," in *IEEE Conference on Computer Communications (INFOCOM)*, Barcelona, Spain, 6–12 May 2007, pp. 1919–1927.

[38] Y.X. Wang, and F.S. Bao, "An entropy-based weighted clustering algorithm and its optimization for ad hoc networks," in *IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, White Plains, NY, USA, October 7–10, 2007, pp. 56–56.

[39] R.S. Bali, N. Kumar, and J.J. Rodrigues, "Clustering in vehicular ad hoc networks: Taxonomy, challenges and solutions," *Vehicular Communications*, vol. 1, no. 3, pp. 134–152, Jul. 2014.

[40] M. Chatterjee, S.K. Das, and D. Turgut, "WCA: A weighted clustering algorithm for mobile ad hoc networks," *Cluster Computing*, vol. 5, no. 2, pp. 193–204, 2002.

Haitao Zhao (Member, IEEE) received the M.S. and Ph.D. (Hons.) degrees in signal and information processing from the Nanjing University of Posts and Telecommunications, Nanjing, China, in 2008 and 2011, respectively. He is currently an Associate Professor with the Nanjing University of Posts and Telecommunications. His current research interests include wireless multimedia modeling, capacity prediction, and wireless network coding.

Jiawen Tang is currently pursuing the master's degree with the College of Telecommunications and Information Engineering, Nanjing University of Posts and telecommunications, Nanjing, China. His research interests include Internet of Vehicles and edge computing, clustering and wireless caching.

Bamidele Adebisi (Senior Member, IEEE) received his Bachelor's degree in electrical engineering from Ahmadu Bello University Zaria, Nigeria, in 1999, and his Masters degree in advanced mobile communication engineering and Ph.D. degree in communication systems from Lancaster University, United Kingdom, in 2003 and 2009, respectively. He was a senior research associate with the School of Computing and Communication, Lancaster University, from 2005 to 2012. He joined Manchester Metropolitan University in 2012, where he is currently a Full Professor (Chair) of intelligent infrastructure systems. He is the current Vice Chair of IEEE TC-PLC; was General Chair, IEEE ISPLC'18, UK; Co-Chair, 6th IEEE Int'l Conference on Smart Grid Communications, 2015, Miami, US, etc. He has published over 140 peer-review papers and given several talks/panel discussions in the research areas of Internet of Things, smart cities, smart grids, communication systems and cyber physical systems. Bamidele is a Fellow of IET, a Fellow of Higher Education Academy and a Chartered Engineer.

Tomoaki Ohtsuki (Senior Member, IEEE) received the B.E., M.E., and Ph. D. degrees in Electrical Engineering from Keio University, Yokohama, Japan in 1990, 1992, and 1994, respectively. From 1994 to 1995 he was a Post Doctoral Fellow and a Visiting Researcher in Electrical Engineering at Keio University. He served as a Chair of IEEE Communications Society, Signal Processing for Communications and Electronics Technical Committee. He has served as general-co chair, symposium co-chair, and TPC co-chair of many conferences, including IEEE GLOBECOM 2008, SPC, IEEE ICC 2011, CTS, IEEE GLOBECOM 2012, SPC, IEEE ICC 2020, SPC, IEEE APWCS, IEEE SPAWC, and IEEE VTC. He gave tutorials and keynote speeches at many international conferences including IEEE VTC, IEEE PIMRC, IEEE WCNC, and so on. He was Vice President and President of the Communications Society of the IEICE. He is a senior member and a distinguished lecturer of the IEEE, a fellow of the IEICE, and a member of the Engineering Academy of Japan.

Guan Gui (Senior Member, IEEE) received the Ph.D. degree from the University of Electronic Science and Technology of China, Chengdu, China, in 2012. Since 2015, he has been a professor with Nanjing University of Posts and Telecommunications, Nanjing, China. His recent research interests include artificial intelligence, deep learning, non-orthogonal multiple access, wireless power transfer, and physical layer security.

Hongbo Zhu received the B.S. degree in communications engineering from the Nanjing University of Posts and Telecommunications (NJUPT), Nanjing, China, in 1982, and the Ph.D. degree in information and communications engineering from the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 1996. He is currently a Professor with the Nanjing University of Posts and Telecommunications (NJUPT). He has authored and coauthored more than 300 technical papers published in various journals and conferences. He is currently Leading a big research group and multiple funds on IoT and wireless communications with current focus on architecture and enabling technologies for Internet of Things. His research interests include Internet of Things, mobile communications and wireless communication theory.