Resource Allocation Algorithms of Vehicle Networks with Stackelberg Game

Ying Zhang[1], Guang-Shun Li[1,2], Jun-Hua Wu[1](✉), Jia-He Yan[1], and Xiao-Fei Sheng[1]

[1] School of Information Science and Engineering, Qufu Normal University,
Rizhao 276800, China
`shdwjh@163.com`

[2] Department of Computer, The Hong Kong Polytechnic University, Hung Hom, Hong Kong

Abstract. With the emergence and development of the Internet of Vehicles (IoV), higher demands are placed on the response speed and ultra-low delay of the vehicle. Cloud computing services are not friendly to reducing latency and response time. Mobile Edge Computing (MEC) is a promising solution to this problem. In this paper, we introduce MEC into the IoV to propose a specific vehicle edge resource management framework, which consists of fog nodes (FN), data service agents (DSA), and cars. We proposed a dynamic service area partitioning algorithm that enables the DSA to adjust the service area and provide a more efficient service for the vehicle. A resource allocation framework based on Stackelberg game model is proposed to analyze the pricing problem of FN and data resource strategy of DSA. We use the distributed iterative algorithm to solve the problem of game equilibrium. Our proposed resource management framework is finally verified by numerical results, which show that the allocation efficiency of FN resources among the cars is ensured, and we also get a subgame perfect nash equilibrium.

Keywords: Internet of Vehicles · Edge computing · Stackelberg game · Wireless resource allocation · Nash equilibrium

1 Introduction

The Internet of Vehicles technology has gradually come into people's field of vision. Many applications in this field, such as safe driving analysis, camera image processing, and optimal route planning, etc., have high requirements for low latency and high throughput [1]. In cloud computing, data service agents aggregate resources into a shared pool, and data service subscribers who need these services can get data services based on demand [2, 3]. However, the geographically long distance between the cloud server and the terminal may cause large network delays and increase costs, which is a great disadvantage for applications that require fast response time and high mobility requirements [4, 5]. Therefore, it is very important to bring resources closer to the end user. In this case, a new architecture and technology called MEC emerged [6]. Therefore, FN is slowly entering people's sight. FNs are lower in the network topology, have a smaller

X. Zhang et al. (Eds.): CloudComp 2019/SmartGift 2019, LNICST 322, pp. 221–230, 2020.
https://doi.org/10.1007/978-3-030-48513-9_18

network delay, and have a wide geographical distribution. These characteristics can meet the small delay and location-aware services of the car as an terminal user requirements for data services [7, 8].

In this paper, MEC technology is introduced into IoV to form vehicular edge computing (VEC). Vehicles cannot directly interface with FNs, so it is important to add a layer of DSA between them. DSA can serve as an intermediary to provide FN data services to the vehicle. When service requests are received from all cars, each DSA can aggregate resources from the FN and sell data services to the car. In this way, cars can use the computing resource services provided by FN through DSA. Each car is a data service subscriber and needs to apply for data services from DSA. We focus on the service radius of each DSA and the benefits of each layer, proposing an efficient data resource management framework.

2 Related Work

In some literatures, fog computing has been proposed as an extension of cloud computing to make up for shortcomings in cloud computing. Ahlgren et al. [9] studied that fog can make up for the shortcomings of the cloud, and cooperate with the cloud to work together. Yannuzzi et al. [10] started from the application scenario of the Internet of Things. The resource limitation of the terminal device talked about the demand for the cloud, and the limitation caused by the position of the cloud in the network talked about the fog. The contrast of clouds and fog, the combination of clouds and fog, the advantages of fog, the application of fog, and the challenges of fog are discussed. Taleb et al. [11] have studied the application of combining edge computing network and cloud computing network. The overview of MEC architecture, typical technologies and main application scenarios are described. In the previous work of our team, we have also studied the resource scheduling in edge computing. Li et al. [12] aimed at the difficulties of resource scheduling in current edge computing, and obtained a new resource scheduling method using improved clustering methods. And in [13], fuzzy clustering and particle swarm optimization are combined to generate a resource scheduling algorithm, which improves the service level and average response time.

Kumar et al. [14] have discussed the advantages of edge computing in building a low-latency, lightweight, high-performance, and highly reliable smart grid platform. It introduces various application scenarios and corresponding edge computing solutions in smart grid. Hou et al. [15] have presented to mine a large number of unused vehicle resources and reduce communication costs. Zhang et al. considered the shortcomings in cloud computing and the massive amount of data, combined with the demand for services in the Internet of Vehicles, transferred some services to the edge computing layer, and proposed a smart vehicle network based on collaborative fog computing in [16]. Li et al. [17] added a service intermediary between the cloud layer and the user, and leased the cloud service to the user. And improved the traditional pricing strategy and realized higher benefits. In [18], a Stackelberg game theory model for bandwidth allocation was established to maximize the benefits of both players through a defined pricing mechanism. Wang et al. [19] analyzed two customer scenarios, these two scenarios are homogeneous and heterogeneous, and built a Stackelberg game based on customer scenarios to improve profits.

3 System Framework

The car needs data service during the driving process, each car can deliver it to the FN at the edge of the network. Each DSA selects a FN to provide cars with the required data services, as shown in Fig. 1. Such a three-layer edge network is the main core framework of this paper. DSA is located in the middle layer, which serves the lower car and manages the upper FN through connecting car and FN. We set the unit of computing resources of each FN to CRB and provide services at a speed of u. Cars rent FN's computing resources through DSA.

The system architecture is shown in Fig. 1. FN stands for fog server, DSA is multi-data agent, and car stands for data service subscribers.

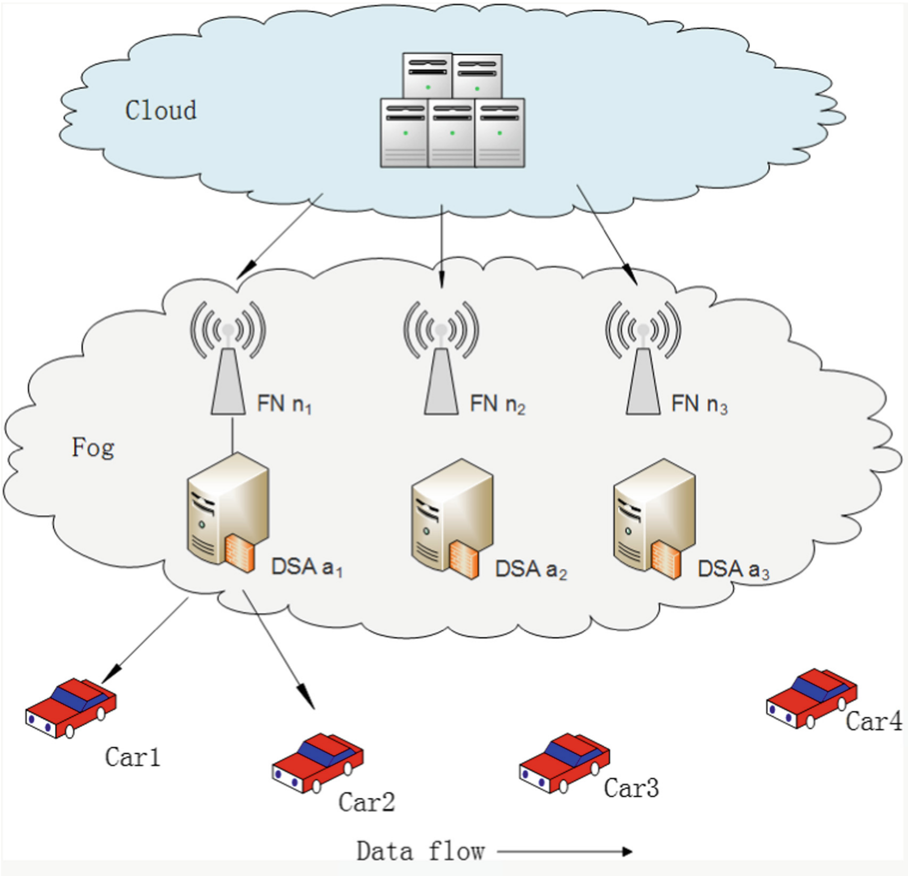

Fig. 1. System framework graph

4 Service Area Partitioning Algorithm

In order to enable the car to apply for services at any time and reduce the service response time, it is necessary to divide the service area of each DSA. The service area of DSA is a circle with a radius of R_{ref}. By default, all cars in this circle are served by the corresponding DSA. Each car can communicate with FN through DSA, or with vehicles in different service areas through mmWare or other networks. One FN is responsible for one DSA, and each DSA can communicate directly with the corresponding FN. Because the traffic conditions on different roads are different and the driving of vehicles is random, the density of vehicles in the service area of each DSA is different, which leads to the unbalanced load of DSA. If there are too many cars requesting data services within a DSA service area, the response time of service will be too long, and even the quality of service will be affected. Therefore, we proposed a dynamic service area partitioning algorithm that helps us to adjust the service area radius R_{ref} according to the car driving behavior in the circle. We mainly consider three factors: speed factor, location factor, server idle resources with the algorithm. Each car mmWare, and each node shares its own information with hello messages.

4.1 Affecting Factors of Service Radius

Speed Factor (VF). Calculated $VF(s, m)$ according to the time interpolation method:

$$VF(s,m) = \frac{|v(m)| - \min_{y \in N_S} |v(y)|}{|v(y)|} \tag{1}$$

Where N_S represents a group of nodes in the neighborhood, $v()$ represents the speed. A smaller VF means small velocity. Based on the weighted exponential moving average, the VF will be updated periodically at an interval of 10 s.

Location Factor (LF)

$$LF(s,i) = \left| tv_{si} - 2(R_{ref\,1} + R_{ref\,2} + \cdots + R_{ref\,\mathrm{i}}) \right| \tag{2}$$

LF indicates the position of the car from the edge of the current service area, tv_{si} indicates the distance traveled by the car s in the area i. And a smaller LF indicates the car is about to leave the service area.

Server Idle Resources

$$\begin{gathered}(1-\delta_i) \leq \delta_{\Delta} \\ s.t \\ \sum_{i=1}^{R} 2\left(R_{ref\,\mathrm{i}}\right)\pi \geq M\,. \\ \exists\, i, \forall\, \mathrm{R} \\ (1-\delta_i) \geq \delta_{\min}\end{gathered} \tag{3}$$

δ_i represents the resource occupancy of the server. All service areas add up to exceed M, and idle resources should be greater than the minimum value. Only by guaranteeing these two conditions can the server provide data resource services.

5 Stackelberg Game Decision

5.1 Stackelberg Game Analysis for Two-Layer Interaction

Assume that in a particular edge computing network, there are multiple FNs, set to set M = {1, 2, …, m}. The set of DSAs covered in this range is N = {1, 2, …, n}, and FN competes for all DSAs. The price strategy of the FN_j node is p_j, exist $j \in M$, and the price strategy of all FNs is $p = (p_1, p_2, \ldots, p_m)$ [20]. The CRB requirement strategy of DSA_i is x_{ij}, exist $i \in N$ which means the quantity of CRB purchased by DSA_i at FN_j. We define $x_i = \left(x_{ij}, x_{-ij}\right)$ as the CRB requirement strategy vector of DSA_i, where x_{-ij} represents the strategy of DSA_i at other FN except FN_j, and $q = (q_1, q_2, \ldots, q_n)$ denotes the set of data requirement strategies for all DSAs.

The game between FN and DSA consists of two phases. In the first phase, different FNs first declare their own price strategy p and broadcast the strategy to all DSAs. In the second phase, the DSA makes its own data resource strategy q based on the received price strategy vector p. The strategic combination of DSA and FN (p, q) is a solution to the Stackelberg game. Next, we formulate the utility of DSA and FN.

The Utility of DSA. The utility function of DSA is composed of two parts: the benefits and costs of providing data services for car. The utility of DSA can be expressed by the following formula:

$$U_{Ni} = U_i\left(\sum_{j=1}^{m} x_{ij}\right) - \sum_{j=1}^{m} P_j\left(p_j x_{ij}\right) - \sum_{j=1}^{m} D_j\left(x_{ij}\right) \tag{4}$$

Where $U_i\left(\sum_{j=1}^{m} x_{ij}\right)$ is the total revenue earned by DSA when car is served by DSA, p_j is the price set by FN, $P_j\left(p_j x_{ij}\right)$ is the price paid by DSA to FN, and $D_j\left(x_{ij}\right)$ represents the delay cost of DSA's service to cars.

We set that the workload of each DSA follows the Poisson arrival process. If the total load $Q_j \geq C_j$ of all DSAs in an FN, the network will be congested. Only when the load of all DSAs in FN satisfies $Q_j < C_j$, the effective transmission of data can be guaranteed. Specifically, the load of FN is Q_j, and the delay cost function of DSA_i in FN_j can be expressed as:

$$D_j(q) = \begin{cases} \frac{\beta_j}{C_j - Q_j}, & \text{if} \quad Q_j < C_j \\ \infty, & \text{if} \quad Q_j \geq C_j \end{cases} \tag{5}$$

Where β_j is a constant related to data transfer technology.

The Utility of FN. For FN, the total utility is total income minus total expenses. We set c_{ij} to the transmission cost per unit CRB, and DSA_i is the service price per unit for FN. Therefore, the utility of FN can be expressed by the following formula:

$$U_{Mj} = \sum_{i=1}^{n} \left(p_j - c_{ij}\right) x_{ij} \tag{6}$$

Where $\sum_{i=1}^{n} p_j x_{ij}$ is the total revenue received by the FN from the DSA, and $\sum_{i=1}^{n} c_{ij} x_{ij}$ is the total transmission cost estimated by the FN.

To get data services from FN, DSA should buy less CRB from FN to get satisfactory service, which comes at a price. How much resources does car need to apply for to complete its own tasks, but also to ensure the quality of service without wasting resources. FN provides data computing services to DSA and needs to give a price that will bring benefits to itself. But if given a higher price, DSA will reduce the number of CRB purchases or chooses another FN, so it is necessary to predict the response of DSA to determine the service price in order to maximize utility. Therefore, how FN price its own resources to protect its own revenue without losing user satisfaction is a key.

Aiming at the characteristics of local sharing of decision information among players in Stackelberg game model mentioned above, we solve the perfect Nash equilibrium (x^*, p^*) of the sub-game with the distributed iteration algorithm proposed in [21].

6 Simulation Experiments

The scene we simulated was in a 3,000-m road. All the cars were running in one direction. The initial DSA service area was 500 m. In this 3,000-m area, we allocated 3 FNs and DSAs. We assume that each car's sensor is in the same location on the car, the rate of data transmission is 50 km/ms, and the delay tolerance of car is 60 ms. In the experiment of iterative algorithm, we only use two service areas covered by different FNs. There are three kinds of DSAs in this area, which are $\alpha = 0.5$, $\alpha = 0.9$ and $\alpha = 2$. Assume that under the initial conditions, the price strategies of both FNs are both 0.1, the value of β is 1, and the initial data resource strategy of the DSA at both FNs is 0.

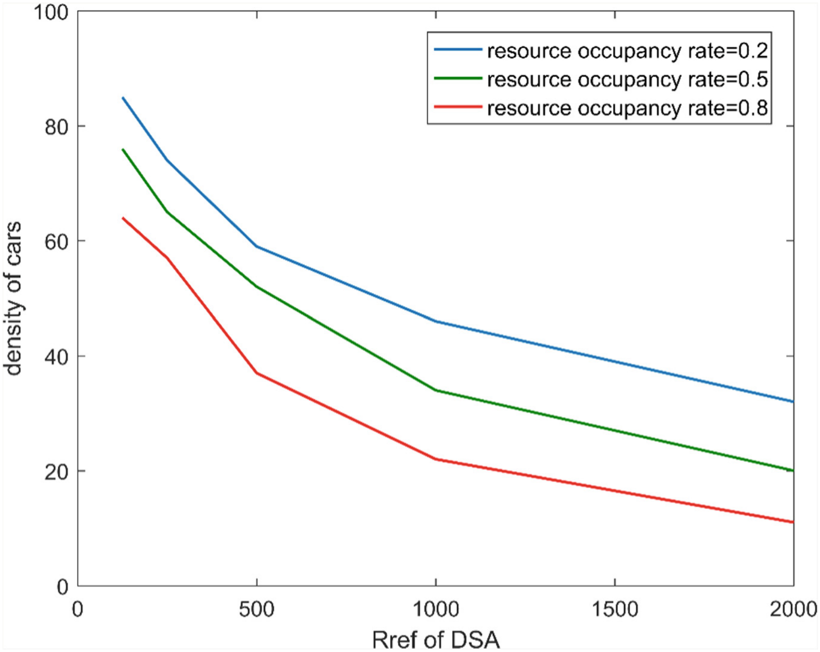

Fig. 2. Relationship between service radius and vehicle density

As shown in the above figure, as the density of vehicles in the area continues to increase, the service area of the DSA will also increase. When the density of car is fixed, we find that the DSA has a higher resource occupancy rate, that is, when the server is busy, the service area of the DSA is smaller. As can be seen from the Fig. 2, the greater the density of car, the less resources available to the server and the smaller the service area.

Figure 3 shows the curve changes of three different utility functions of DSA in the iteration process. The users of curve $\alpha = 2$ are cars which is the least sensitive to data service delay, so the corresponding price paid by car is relatively low. The resource in FN is preferentially contested by DSA which is sensitive to delay and applies for more resources, so the effect is relatively low. The curve of $\alpha = 0.5$ is car is the most sensitive to data service delay, so DSA needs to buy more data resources from FN, price strategy of FN will be more friendly, so the utility of DSA is higher, but as the equilibrium point of game is reached, the resource of FN is effectively utilized, and the utility of DSA decreases accordingly. Car in the DSA service area of $\alpha = 0.9$ is moderately sensitive to service delay, and the curve increases gradually at first, and then gradually becomes stable. All three curves tend to stabilize after reaching that equilibrium point.

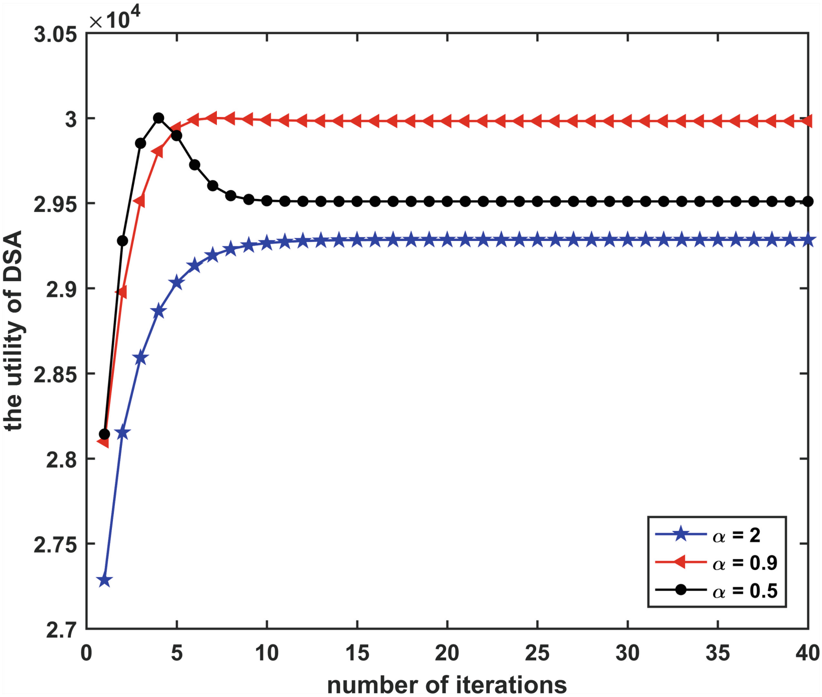

Fig. 3. Different user utility changes with the iterative process

From Fig. 4, we obtain the subgame perfect Nash equilibrium of the Stackelberg game of heterogeneous wireless networks. The two curves in the figure are the optimal price curve of the first FN and the optimal price curve of the second FN. The intersection of the two curves is the Nash equilibrium point p^*. The intersection of two curves corresponds to the optimal price strategy of two FNs. At this point, both the FN layer and the DSA layer have reached the Nash equilibrium, and the subgame perfect Nash equilibrium is obtained for the Stackelberg game.

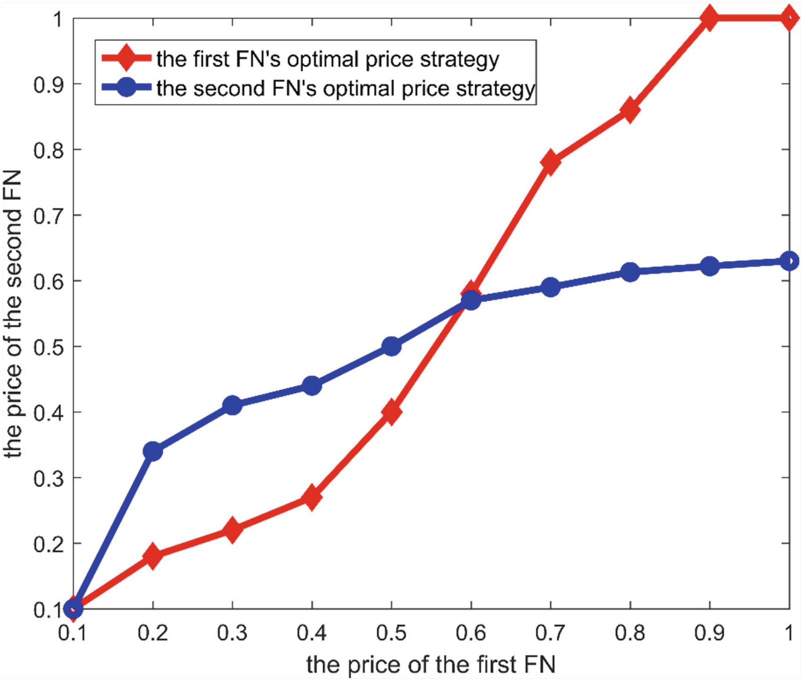

Fig. 4. Nash equilibrium of game between FNs

7 Conclusion

In this paper, we propose a framework for FN, DSA and car scenarios for VEC. Under this framework, we first judge the service area of the DSA according to the characteristics of the car movement. On this basis, a Stackelberg game model is established to study the pricing of FN and the optimal purchase amount of DSA resources. At each stage, it is necessary to meet the best results of the participants, and no one in this framework can unilaterally change their behavior to achieve better utility. Through simulation results, it is proved that both sides of the game, FN and DSA, have reached the highest effectiveness, and the proposed framework achieves high-performance services. For future work, we can consider the contact between the cars of lowest layer, and whether some data services can be obtained in the neighborhood vehicles. If data services can be provided between cars, how to choose neighbor vehicles to provide services to achieve lower latency services.

References

1. Hu, Q., Wu, C., Zhao, X., Chen, X., Yoshinaga, T.: Vehicular multi-access edge computing with licensed Sub-6 GHz, IEEE 802.11p and mmWave. IEEE Access **6**, 1 (2017)
2. Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M.: Internet of Things (IoT): a vision, architectural elements, and future directions. Future Gener. Comput. Syst. **29**(7), 1645–1660 (2013)

3. Zhang, H., Yong, X., Bu, S., Niyato, D., Yu, R., Zhu, H.: Computing in resource allocation three-tier IoT fog networks: a joint optimization approach combining stackelberg game and matching. IEEE Internet Things J. **4**(5), 1204–1215 (2017)
4. Dastjerdi, A.V., Buyya, R.: Fog computing: helping the Internet of Things realize its potential. Computer **49**, 112–116 (2016)
5. Ning, Z., Kong, X., Xia, F., Hou, W., Wang, X.: Green and sustainable cloud of things: enabling collaborative edge computing. IEEE Commun. Mag. **57**, 72–78 (2019)
6. Greenberg, A.G., Hamilton, J.R., Maltz, D.A., Patel, P.: The cost of a cloud: research problems in data center networks. ACM SIGCOMM Comput. Commun. Rev. **39**, 68–73 (2008)
7. Goiri, I., Le, K., Guitart, J., Torres, J., Bianchini, R.: Intelligent placement of datacenters for internet services. In: 2011 31st International Conference on Distributed Computing Systems, pp. 131–142 (2011)
8. Yu, L., Chen, L., Cai, Z., Shen, H., Liang, Y., Pan, Y.: Stochastic load balancing for virtual resource management in datacenters. IEEE Trans. Cloud Comput. **PP(99)**, 1 (2016)
9. Ahlgren, B., et al.: Content, connectivity, and cloud: ingredients for the network of the future. Commun. Mag. IEEE **49**(7), 62–70 (2011)
10. Yannuzzi, M., Milito, R.A., Serral-Gracià, R., Montero, D., Nemirovsky, M.: Keyingredient sinan iotrecipe: fog computing, cloud computing, and more fog Computing. In: IEEE International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (2015)
11. Taleb, T., Samdanis, K., Mada, B., Flinck, H., Sabella, D.: On multi-access edge computing: a survey of the emerging 5G network edge architecture and orchestration. IEEE Commun. Surv. Tutor. **PP**(99), 1 (2017)
12. Li, G., Xu, S., Wu, J., Ding, H.: Resource scheduling based on improved spectral clustering algorithm in edge computing. Sci. Program. **2018**(5), 1–13 (2018)
13. Li, G., Liu, Y., Wu, J., Lin, D., Zhao, S.: Methods of resource scheduling based on optimized fuzzy clustering in fog computing. Sensors **19**(2), 2122 (2019)
14. Kumar, N., Zeadally, S., Rodrigues, J.J.P.C.: Vehicular delay-tolerant networks for smart grid data management using mobile edge computing. IEEE Commun. Mag. **54**(10), 60–66 (2016)
15. Hou, X., Yong, L., Min, C., Di, W., Jin, D., Sheng, C.: Vehicular fog computing: a viewpoint of vehicles as the infrastructures. IEEE Trans. Veh. Technol. **65**(6), 3860–3873 (2016)
16. Zhang, W., Zhang, Z., Chao, H.C.: Cooperative fog computing for dealing with big data in the internet of vehicles: architecture and hierarchical resource management. IEEE Commun. Mag. **55**(12), 60–67 (2017)
17. He, L., Dong, M., Ota, K., Guo, M.: Pricing and repurchasing for big data processing in multi-clouds. IEEE Trans. Emerg. Top. Comput. **4**(2), 1 (2016)
18. Cong, W., Ying, Y., Wang, C., Xi, H., Zheng, C.: Virtual bandwidth allocation game in data centers. In: IEEE International Conference on Information Science and Technology (2012)
19. Hao, W., Zhao, Y., Guan, H.: On pricing schemes in data center network with game theoretic approach. In: International Conference on Computer Communication and Networks (2014)
20. Pu-yan, N., Pei-ai, Z.: A note on Stackelberg games. In: 2008 Chinese Control and Decision Conference, pp. 1201–1203 (2008)
21. Jiang, Y., Chen, S.Z., Hu, B.: Stackelberg games-based distributed algorithm of pricing and resource allocation in heterogeneous wireless networks. J. Commun. **1**, 61–68 (2013)