Cooperative Fog Computing for Dealing with Big Data in the Internet of Vehicles: Architecture and Hierarchical Resource Management

Wenyu Zhang, Zhenjiang Zhang, and Han-Chieh Chao

By considering factors like latency, mobility, localization, and scalability, the authors propose a regional cooperative fog computing based intelligent vehicular network (CFC-IoV) architecture for dealing with big IoV data in the smart city. Possible services for IoV applications are discussed, including mobility control, multi-source data acquisition, distributed computation and storage, and multi-path data transmission.

ABSTRACT

As vehicle applications, mobile devices and the Internet of Things are growing fast, and developing an efficient architecture to deal with the big data in the Internet of Vehicles (IoV) has been an important concern for the future smart city. To overcome the inherent defect of centralized data processing in cloud computing, fog computing has been proposed by offloading computation tasks to local fog servers (LFSs). By considering factors like latency, mobility, localization, and scalability, this article proposes a regional cooperative fog-computing-based intelligent vehicular network (CFC-IoV) architecture for dealing with big IoV data in the smart city. Possible services for IoV applications are discussed, including mobility control, multi-source data acquisition, distributed computation and storage, and multi-path data transmission. A hierarchical model with intra-fog and inter-fog resource management is presented, and energy efficiency and packet dropping rates of LFSs in CFC-IoV are optimized.

INTRODUCTION

The intelligent transportation system (ITS) is a basic essential for sustaining a smart city, and the Internet of Vehicles (IoV) plays an important role in ITS for its advantages in reducing traffic accidents, alleviating traffic congestion, and providing various real-time convenience services, such as navigation, multimedia entertainment, and remote fault diagnosis [1]. With the rapid growth of vehicle applications, mobile devices, and the Internet of Things (IoT) in IoV, how to deal with big transportation data is still an open problem in IoV.

To offload the computation burden of resource-constrained vehicles, cloud computing has been adopted by gathering the computation tasks to central cloud servers, and the results are sent back to local users. For example, in some navigation applications, the GPS data of vehicles are dynamically uploaded to the cloud server, in which the congestion states of the streets in the city can be obtained; thus, congested route paths with shortest distance will be abandoned by users. However, IoV is a heterogeneous network consisting of ad hoc vehicle networks, IoT, roadside networks, and cellular mobile networks. Exploiting a central cloud mode for big data management in IoV still faces the following drawbacks.

Latency: Real-time data analysis is a stringent requirement in the highly dynamic IoV environment. However, in centralized cloud computing, services are centrally provisioned in the cloud center, and the big data between cloud centers and IoV need to be transmitted through the wide area network (WAN); hence, the delay or delay-jitter is unavoidable.

Mobility support: In IoV, vehicles are always in mobile state. However, mobility support for IoV has not yet been considered in conventional centralized cloud computing frameworks [2]. The relatively high delay makes it inefficient to use remote cloud servers to provide mobility control services.

Scalability: The IoV is composed of vehicular ad hoc networks (VANETs), mobile networks, and roadside IoT. Providing solutions to support these heterogeneous networks is not considered in conventional cloud computing [3].

Efficiency: Millions of mobile devices, vehicles, and sensors bring a huge burden for the cloud center to dynamically process the big data of IoV; therefore, efficiency is also a problem in centralized processing environments.

The root cause of the above problems is the centralized computation and storage of cloud services. Thus, some improved hierarchical paradigms have been proposed, such as mobile cloud [4], and cloudlet [5], to facilitate the localization of cloud services while at the same time retaining the dominance of the central cloud server. In a similar way, fog computing provides a new paradigm to offload computation from cloud to local fog servers (LFSs) [6], and a few works have proposed to adopt fog computing in IoV. In [7], a local roadside cloud-based network is proposed to deal with traffic-related data, which is coincident to the goal of fog computing. A vehicular fog computing (VFC) architecture is proposed in [8], in which vehicles with redundant resources are used as the computational infrastructures, and the burdens of congested resource-limited vehicles are relieved. In [9], the software defined network (SDN) is introduced as the basic routing protocol for fog-computing-based IoV. The SDN controller is also introduced for resource

Digital Object Identifier:
10.1109/MCOM.2017.1700208

Wenyu Zhang and Zhenjiang Zhang are with Beijing Jiaotong University; Han-Chieh Chao is with National Ilan University.

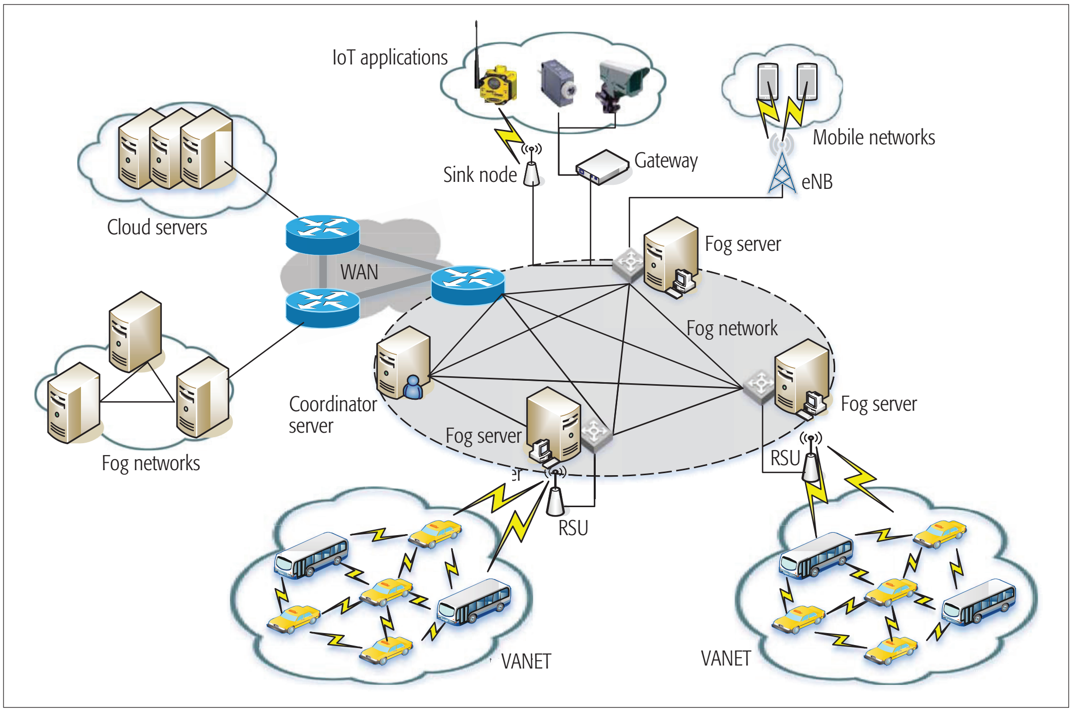

Figure 1. Proposed cooperative fog architecture for the Internet of Vehicles.

The proposed CFC-IoV architecture is mainly composed of two layers: fog layer and edge layer. The fog layer is a federation of geographically distributed LFSs, coordinator server, and cloud servers. The edge layer includes the VANET, IoT applications, and mobile cellular networks, which are essential elements in supporting IoV applications.

management tasks. The rule installation problem of SDN-based IoV is studied in [10], and a destination-driven model is proposed to improve the installation efficiency at switches.

The hierarchical cloud-based or SDN-based framework is suitable for large wide area networks (WANs). In this article, we emphasize the principal role of fog servers, and present a regional cooperative fog computing (CFC) architecture to support IoV applications. Different from controlling by a cloud center, a localized coordinator is introduced to provide interoperability and cooperative operation between LFSs. Possible services dealing with big IoV data in CFC-IoV are discussed, including mobility support and service migration, multi-source data acquisition, distributed processing and storage, and multi-path data transmission. We then develop a hierarchical resource management model for performance optimization in the proposed fog network. Energy efficiency is optimized in intra-fog resource management, and the dropping rate of congested LFSs are minimized in inter-fog resource management.

The remainder of this article is organized as follows. We first illustrate the CFC-IoV architecture. Next, cooperative services in CFC-IoV are discussed. Subsequently, task classification and resource management for CFC is devised. The analyzed model of hierarchical resource management problem is developed, along with some simulation results. We finally conclude this article.

Proposed Architecture

Currently, the IEEE 820.11p standard, also called Wireless Access in Vehicular Environments (WAVE) [11], has been released to use 5.9 GHz to support short-range wireless communication in vehicular networks, including vehicle-to-vehicle (V2V) communication and vehicle-to-roadside (V2R) communication. Other techniques, such as WiFi, device-to-device (D2D) communication, and 3G/4G LTE, also provide alternatives for enabling wireless communication in IoV. The V2V communication is usually realized in a VANET, a protocol for sharing and gathering data between vehicles, and it has been studied extensively in recent years [12]. On the other hand, IoV is an organic combined system with the integration of VANETs and roadside infrastructures, including roadside units (RSUs), traffic surveillance systems, mobile networks, and the Internet, which are connected through the V2R communication.

As shown in Fig 1, the proposed CFC-IoV architecture is mainly composed of two layers: the fog layer and edge layer. The fog layer is a federation of geographically distributed LFSs, a coordinator server, and cloud servers. The edge layer includes the VANET, IoT applications, and mobile cellular networks, which are essential elements in supporting IoV applications. In addition to advantages like low latency, mobility support, and efficiency, the proposed CFC-IoV architecture also has the following distinctive features.

Open: An intelligent IoV is not able to gain sufficient information on traffic conditions without the assistance of IoT applications and mobile networks. Contrary to the closed mobile cellular networks, CFC-IoV is an open system to heterogeneous networks. Authorized users, such as government, service providers, and users, are able to add new entities to the fog computing system. Various access ways, including evolved NodeBs (eNBs), WiFi access points (APs), and gateways, can be used to build connections between LFSs and users.

Organized and Autonomous: The core network of CFC-IoV is composed by geographically distributed LFSs. Although LFSs are independent entities, the whole network is coordinated and controlled by a local coordinator server. By default, the IoV edge networks, such as VANETs

One of the important functions of fog computing is the support for mobility and service migration. Similar to the concept of VM migration, when mobile vehicle users move between different LFS domains, coordination mechanisms can be provided to maintain the service uninterrupted.

and surveillance systems, are connected to their nearest LFSs, but it is also possible to be allocated other LFSs with acceptable delays to optimize the performance of the whole system. On the other hand, fog servers are autonomous to complete their own customized applications and intra-fog resource management operations without the control of the cloud center and the coordinator server. In this way, the burden of the coordinator can be alleviated, and configurations of fog servers can be more flexible and practical.

Regional: Different from cloud-based or SDN-based architectures, the proposed architecture is not designed for WANs. To guarantee efficiency and low latency, it is regional, and the coverage area is limited, such as the metropolitan area network of a city. Therefore, it can be regarded as a supplement to cloud computing with larger coverage areas.

Fog Layer

The fog layer may include the following entities:

Fog servers: The computation and storage for big data of the edge layer is offloaded to LFSs; thus, long-haul data transmission between vehicles and the cloud center is avoided. Also, an LFS can provide many more services and works differently than cloud computing, which is illustrated in the next section. Except for the member RSUs, a fog server may be equipped with the following functional entities.

Access control routers (ACRs): To control or migrate the input data flow, access control is essential. When the incoming data flow is too large, packets will be dropped or migrated to other idle LFSs.

Virtual machines (VMs): In a fog server, network functions virtualization (NFV) can be enabled, and the resource is assumed to be divided into several independent VMs.

Adaptive load dispatcher (ALD): An ALD can be used to adjust the data flow of each VM to achieve intra-fog resource management. This practice is actually similar to resource management in virtualized cloud centers.

Cloud servers: In CFC-IoV, the cloud server locates outside the fog layer, but it can be regarded as a "super fog server" with sufficient computation and storage capacities. However, network configuration and control functions will be removed to the coordinator server, which is also deployed in the fog layer.

Coordinator server: We introduce a regional coordinator to guarantee the federation and autonomy of fog networks. The coordinator can be used to deal with the following tasks. Applications with very high computation complexity and massive storage are still completed in cloud centers.

•Inter-fog resource management. The LFSs dynamically update their work to the coordinator server. By scheduling the data flow and adjusting operating conditions of fog servers, resource management can be conducted to guarantee the quality of service (QoS) of users or optimize the performance of member fog servers.

•Security and privacy protection. Unlike mobile cellular networks, the proposed CFC-IoV is an open architecture to service providers, governments, and even individuals. It is important to provide security and privacy protection services for the fog network, and the coordinator server can be the one to take this role.

•High-level information fusion. IoV is composed of distributed, multiple, and heterogeneous subnetworks. Applications like traffic monitoring, navigation, and cognitive radio may produce big data to process. Since data can be processed and stored in LFSs, the coordinator can be used to conduct information fusion operations to obtain high-level decisions, including global and regional traffic conditions, available spectrum pools, and so on.

Edge Layer

The edge layer may include the following components:

VANET. The VANET is useful in applications like traffic information broadcasting and V2V communications. The operations in VANETs are independent of the fog layer. However, resource allocation and interference cancellation operations in V2R wireless communication are controlled by the domain fog server.

IoT. IoT applications, such as video traffic surveillance systems, range finders, RFIDs, and wireless sensors, are widely used in city transportation systems. These applications can be properly integrated into IoVs to provide various kinds of information to assist IoV applications.

Mobile cellular networks. Mobile devices have close relationships with IoV. Many IoV applications, such as navigation, traffic information acquisition, taxis (e.g., Uber and DiDi), and shared bikes, are provided by mobile devices. Also, a cellular eNB can also be used as an RSU to build connections between fog servers and vehicles. Therefore, mobile cellular networks are also essential to IoV.

Cooperative Fog Computing

We emphasize the synergistic cooperation of LFSs for dealing with big IoV data. As shown in Fig. 2, a CFC network can be used to provide the following four cooperative services.

Mobility Support and Service Migration

One of the important functions of fog computing is support for mobility and service migration. Similar to the concept of VM migration, when mobile vehicle users move between different LFS domains, coordination mechanisms can be provided to maintain uninterrupted service.

As shown in Fig. 2a, when a user moves from LFS A to LFS B, LFS A can tell LFS B that the user is coming, and cooperative handover is conducted to keep the service going. This handover process is coordinated by LFSs or a coordinator server, but not the central cloud, as all controlling operations are conducted in the CFC network itself.

Multi-Source Data Acquisition

The information involved in IoV can be geographic map data, traffic monitoring data, spectrum information, available parking lots, available charging stations, and so on. All of these data need to be dynamically updated. To support multi-source data acquisition, each LFS is responsible for the relevant information within their own member edge networks; thus, the burden of the

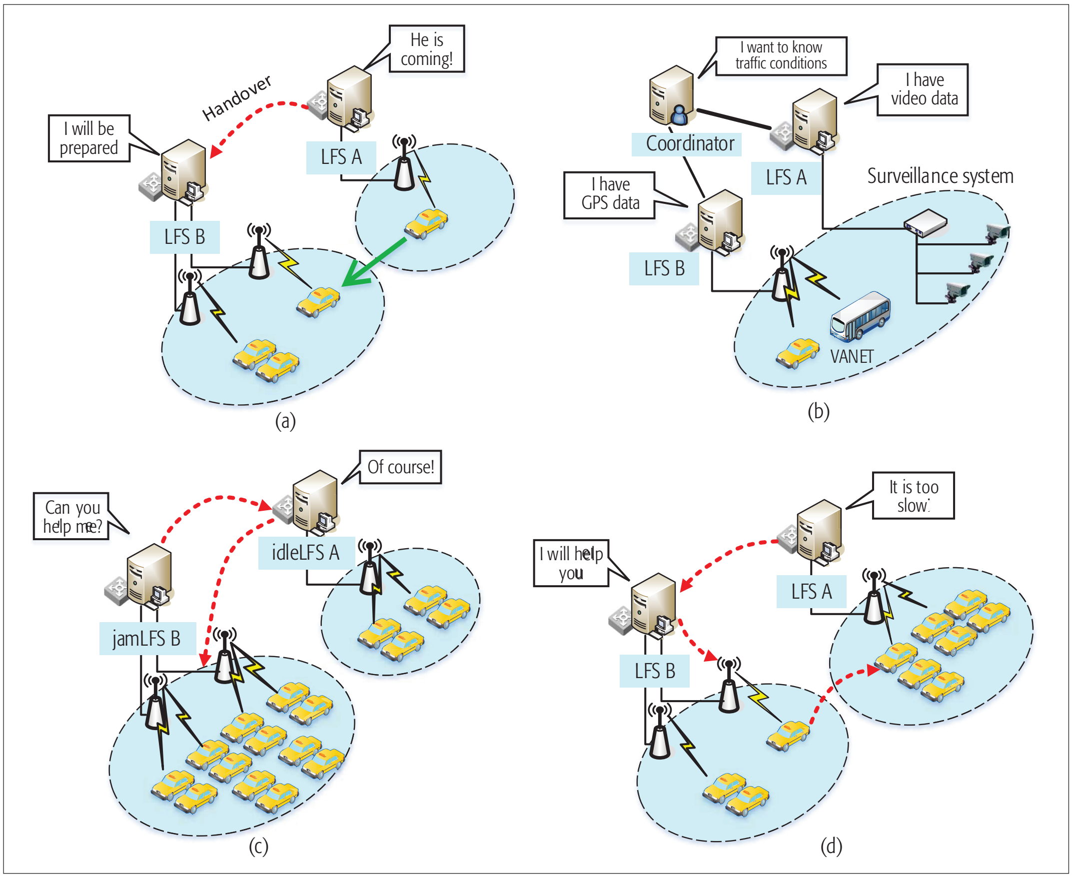

Figure 2. Scenarios of cooperative fog computing: a) cooperative service handover in mobility support; b) multi-source data acquisition in traffic monitoring; c) distributed computation; d) multi-path file downloading.

Supporting communication for millions of mobile devices and sensors is a huge burden for LFSs. Some IoV applications, such as in-vehicle multimedia entertainment, may bring large volume of data stream in fog computing. This problem can be alleviated by using multi-path data transmission techniques.

coordinator server will be greatly reduced. After processing, the refined information can be initially spread to other LFSs, and can also be passively waiting for the queries of other LFSs. The former is suitable for busy conditions, while the latter is suitable for idle conditions.

Figure 2b gives an example of cooperative sensing data acquisition when the coordinator wants to know the traffic conditions of all the streets in the city. LFS A can provide video surveillance data, while LFS B can provide the GPS data of the vehicles. By using cooperative data acquisition, the breadth and depth of obtained data can be improved, and thus better understanding of city traffic will also be obtained.

Distributed Computation and Storage

Distributed data processing and storage are helpful for digesting big IoV data in local LFSs, because transmitting big data to a cloud center or coordinator server can be avoided. Also, the service requests of users can be rapidly fulfilled, and the efficiency of the network can be improved.

On the other hand, the vehicles are not uniformly distributed in a city, and the workloads of LFSs are always unbalanced. To balance the workloads of LFSs, distributed computation and storage can be exploited to conduct service migration operations to relieve the stress of jammed LFSs (jamLFS).

An example scenario is given in Fig. 2c, in which idle LFS (idleLFS) A has a low volume of service requests, while jamLFS B with a heavy workload has too many vehicles to wait for service. In this situation, the jamLFS can allocate some workloads to its neighboring idleLFSs. Thus, the congestion of the jamLFS can be alleviated or solved. As a result, low latency between jamLFS and users can still be preserved.

Multi-Path Data Transmission

Supporting communication for millions of mobile devices and sensors is a huge burden for LFSs. Some IoV applications, such as in-vehicle multimedia entertainment, may bring large volumes of data streams in fog computing. This problem can be alleviated by using multi-path data transmission techniques.

As shown in Fig. 2d, the wireless channel is shared by too many vehicles, and the speed of file downloading is slow. In this situation, its neighbor, LFS B, can be used to build multi-path routing to transmit data via the VANET for LFS B to the user, and the file downloading speed will increase. In a similar way, this practice can be used to help transmit data between LFSs and central clouds.

The potential applications in the proposed CFC-IoV are summarized in Table 1. We can see that cooperation of fog servers is applicable in all the applications. In particular, multi-path data transmission is possible in all potential applications, which means that the services of fog servers can be migrated to other LFSs. Accordingly, resource management should be conducted to optimize the performance of the whole network, and is discussed in the following section.

It can be observed that the energy consumption rate with energy-aware resource allocation is always much lower than in static allocation mode. When using a static allocation policy, energy consumption becomes higher when we have more VMs. In contrast, when energy allocation is adopted, energy expenditure will be lower with more VMs.

Applications	Mobility support and service migration	Multi-source data acquisition	Distributed computation and storage	Multi-path data transmission
Real-time traffic monitoring		✓	✓	✓
Video surveillance		✓	✓	✓
Real-time navigation	✓	✓	✓	✓
Multimedia and in-vehicle entertainment	✓			✓
Cognitive radio	✓	✓		✓
Parking lot and charging station search		✓	✓	✓
Remote vehicle diagnosis	✓		✓	✓

Table 1. Potential applications in fog-computing-based Internet of Vehicles.

Hierarchical Resource Management

The objectives of resource management in CFC-IoV mainly include two aspects:

•Energy efficiency, which is helpful to reduce system maintenance cost and energy consumption. Also, reducing energy consumption is friendly to the city environment. In this article, we mainly focus on energy efficiency optimization for data communication and processing in fog servers and energy consumption by wireless communication in RSUs; vehicles are not considered.

•QoS, which is mainly determined by the packet dropping rate of each LFS. In normal conditions, a fog server is able to meet requests from its own domain. However, in some areas with high population density, or when traffic congestion happens, the corresponding traffic flow may be too high for resource-limited LFSs. Resource management can be conducted by the coordinator to balance the loads by assigning the traffic of a congested LFS to a nearby LFS. When dropping rate is decreased, latency is also avoided.

Achieving the above two objectives at the same time requires the coordinator to control every VM of its member LFSs, which is a huge job with high cost, since the number of VMs can be very large in a fog network, and the corresponding computation complexity is too high. Also, as LFSs are autonomous, they have priority to determine their own working state, and are not dominated by the coordinator server. As such, this article proposes a flexible hierarchical resource management methodology, as shown in Fig. 3, which includes the following.

•Intra-fog resource management. In a virtualized fog server (VFS), the size of a task assigned to each virtual machine (VM) is controlled by an adaptive load dispatcher (ALD), and the processing rate of each VM is also adjustable.

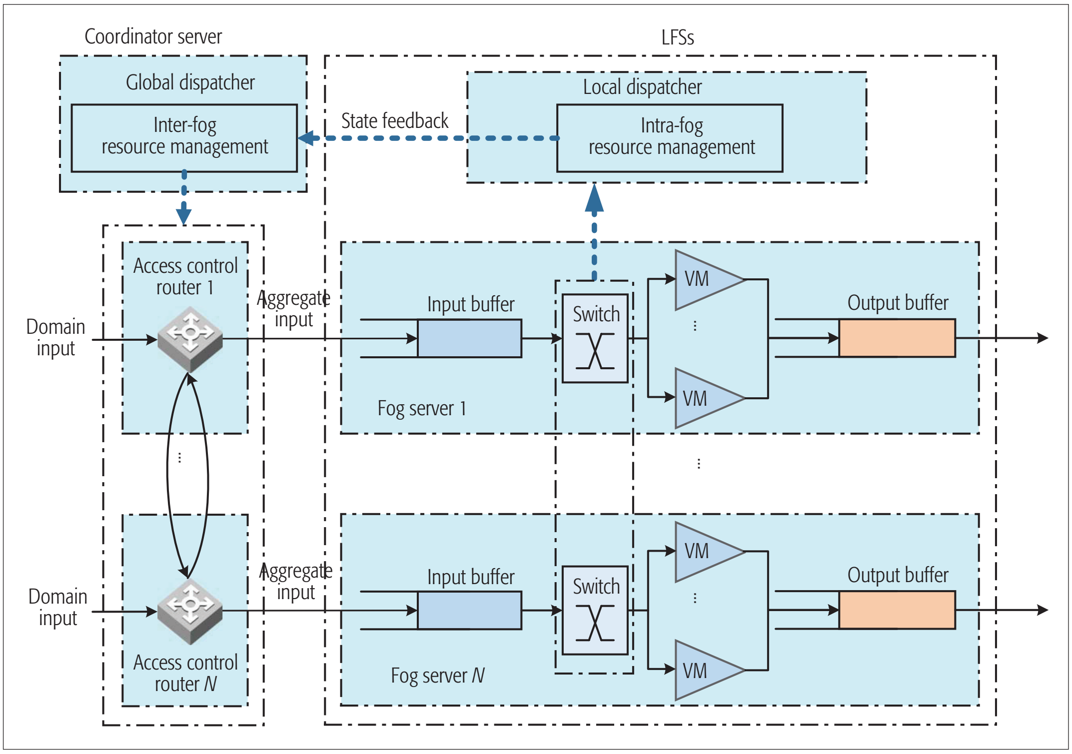

Figure 3. Hierarchical resource management in CFC-IoV.

•Inter-fog resource management. The LFSs report their working state to the coordinator server, which is able to tell the jamLFSs to assign overflowing workload to nearby idleLFSs. This inter-fog resource management operation can be achieved by controlling the data flow in the access control routers, and intra-fog operations remain undisturbed. Cooperating with intra-fog resource management, the latencies of the LFSs can be optimized.

Intra-Fog Resource Management

Suppose that a VFS is equipped with M source-limited VMs v_1, ..., v_M. The size of workload of each VM is dynamically assigned by the ALD, and denote ξ_i as the assigned workload of VM v_i. Let ξ be the total coming workload from the input buffer; thus, we have constraint $\sum_{i=1}^{M} \xi_i = \xi$. Due to the differences of hardware and the external environments, energy consumption rates and computation speeds of the VMs are always different. For v_i, denote its CPU processing rate as u_i, and its maximum processing rate is u_i^{max}. When v_i is in idle state, its energy consumption rate is e_i^{idle}, and when v_i is in full load, its maximum energy consumption rate is e_i^{max}. The energy consumption rate on computation can be estimated by $e_i^{cp} = e_i^{idle} + (u_i/u_i^{max})^2 (e_i^{max} - e_i^{idle})$ [13]. The energy expenditure on data communication from input buffer to VM v_i can be calculated by $e_i^{cm} = \gamma \xi_i^2$, and γ is a constant scaling factor. Also, let z be the transmitting speed of the output buffer; we assume that z is linearly determined by ξ (i.e., $z = \eta \xi$), where η is a positive constant. The energy consumption on draining data from the output buffer can be approximated as $e^{tr} = \rho z^2$, where ρ is a constant scaling factor. More detailed calculation of e_i^{cm} and e^{tr} can be found in [13, 14]. The total energy consumption of a VFS can be calculated by

$$\mathbb{E}(e) = \mathbb{E}\left(\sum_{i=1}^{M} e_i^{cp}\right) + \mathbb{E}\left(\sum_{i=1}^{M} e_i^{cm}\right) + \mathbb{E}(e^{tr}) \qquad (1)$$

where $\mathbb{E}(x)$ denotes the expectation of variable x. A convex optimization model can be obtained by minimizing the expected energy consumption of the fog server while guaranteeing the data processing rate. By constructing the Lagrange dual function and using the Karush-Kuhn-Tucker (KKT) conditions, the optimization problem can be solved.

Figure 4 plots average results on energy consumption rate with different aggregated arrival rates from 2–10 Mb/s. The minimum and maximum energy consumption of all the VMs are different, and range from 0.5–1.5 J/s and 2.5–3.5 J/s, respectively. The maximum processing rate of the VMs range from 3–5 Mb/s. Two working modes are compared: static allocation mode and energy-aware allocation mode. In the former, the assigned data flow and the processing speed of each VM are the same but just meet the requirement of processing requests. In the latter, assigned data flows and processing speeds are dynamically adjusted by using the above optimization model.

It can be observed that the energy consumption rate with energy-aware resource allocation is always much lower than static allocation mode. When using static allocation policy, energy consumption becomes higher when we have more VMs. In contrast, when energy allocation is adopted, energy expenditure will be lower with more VMs. These results highlight the importance of intra-fog resource management in improving the energy efficiency, especially for VFSs with relatively large VMs numbers.

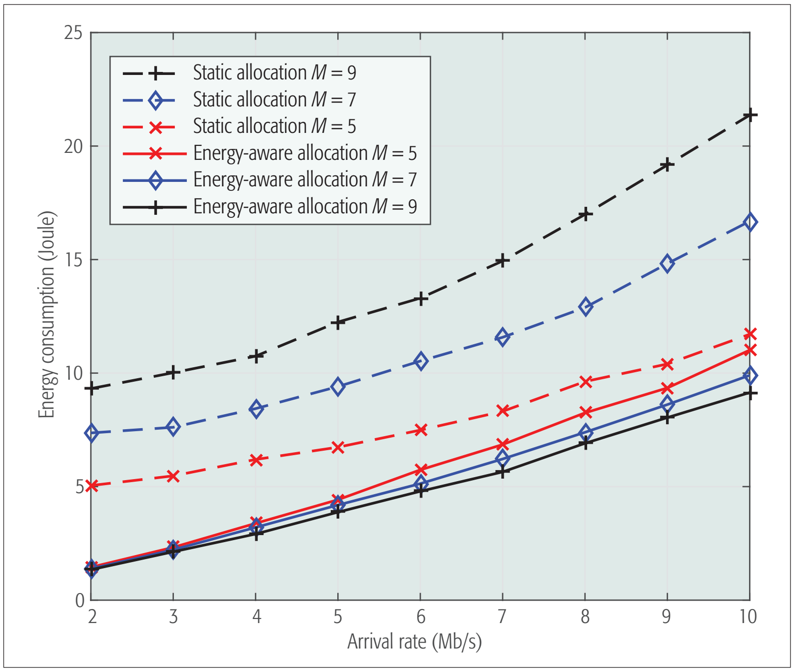

Figure 4. Energy consumption as a function of arrival rate in intra-fog resource allocation.

Inter-Fog Resource Management

On the basis of the cooperation of LFSs, we propose an inter-fog QoS-aware resource management algorithm to improve the stability and load balancing of the fog network. In this mode, the fog servers are regarded as independent entities. The coordinator conducts resource management operations by adjusting the data flow to access control routers of the LFSs, but does not control the operation modes of LFSs, such as intra-fog dispatching policy and VM processing rates. The essential information for resource management, such as input/out buffer size, domain input rate, queen length, and transmit rates of input/output buffer, are known by default by the global coordinator, which can be achieved by periodic uploading from LFSs.

As shown in Fig. 3, consider a CCF-IoV with N LFSs s_1, ···,s_N, whose corresponding buffer sizes are L_1^{max},?,L_N^{max}. The domain input traffic size of LFS s_i is denoted as a_i, and we assume that a_i follows a Poisson process model. Denote the network traffic assignment matrix as **w**, in which w_{ij} denotes the ratio of domain traffic of LFS s_i assigned to LFS s_j. Apparently, we have constraint $\sum_{j=1}^{N} w_{ij} = 1$. When an LFS is not congested, we have $w_{ii} = 1$ and $w_{ij} = 0$, $j \neq i$.

We assume that the links between fog servers are all bidirectional and symmetrical. The transmission delay between the routers of s_i to s_j are denoted as t_{ij}. For a congested LFS s_i, if t_{ij} is too high, s_j is less probable to be selected to help s_i. The optimal traffic assigning probability matrix can be derived by minimizing the maximum delay

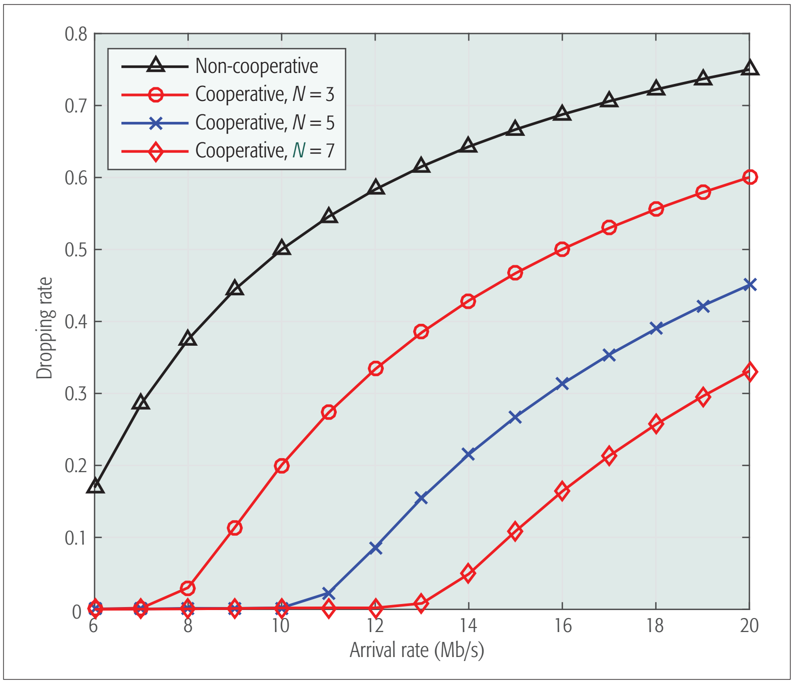

Figure 5. Packet dropping rate as a function of arrival rate in inter-fog resource allocation.

time of all LFSs while migrating the data to nearby LFSs as much as possible, as given by

$$\min_{\mathbf{w}} \max_{1\le i\le N} \sum_{j=1}^{N} w_{ij}\left(\mathbb{E}\left(p_j^{drop}\right)+\beta\mathbb{E}(t_{ij})\right)$$
$$\text{s.t.} \sum_{j=1}^{N} a_j w_{ji} \le \tau_i \qquad (2)$$

where β is a scaling coefficient controlling the impact of migrating delay, and τ_i is the maximum aggregate workload of s_i. Mathematically, one can use the $M/M/1/L_i^{max}$ queening model to calculate the dropping rate of an LFS. However, this is not essential, since it is more practical for the LFSs to directly upload the dropping rates to the coordinator server. The above min-max optimization can be transferred to a constrained min-optimization problem, and KKT conditions also can be applied to find the solutions. In practice, heuristic methods can be used to dynamically adjust the allocation weights according to the state feedback of LFSs, and hence the workload of congested LFSs can be balanced.

We use a CFC-IoV network with different arrival rates to test the performance of inter-fog QoS resource allocation. The buffer size of each LFS ranges from 20–30 Mbits. The maximum processing rate of each LFSs is assumed to be 5 Mb/s. The domain traffic of all LFS ranges from 1–4 Mb/s, except one jamLFS with heavy incoming workloads. Delays between LFSs range from 30–70 ms, and scaling factor β = 0.1. The following two modes are tested: non-cooperative mode and cooperative resource allocation mode. In the former, domain traffic only can be processed by corresponding LFS, while in the latter, domain traffic can be reassigned to other LFSs to alleviate the burden of jamLFS.

The dropping rates when domain arrival rates ranging from 6 to 20 Mb/s of the jamLFS are shown in Fig. 5. Obviously, in the non-cooperative scenario, the dropping rate becomes higher with increasing domain arrival rate, which will cause serious delay between users and jamLFS. The situation gets much better when inter-fog cooperative resource allocation is adopted, which means that cooperative fog computing has strong self-adaptability to deal with imbalance or outburst of data flow. Also, dropping rates become lower when the number of member LFSs becomes larger; thus, a larger cooperative fog network has stronger stability.

When most of the LFSs are congested, the proposed optimization model will have poor efficiency, since no more extra capacity can be used to help the JamLFSs. As aforementioned, the remote cloud servers can be regarded as a powerful fog node. Therefore, when the traffic of the whole network is too high to be digested in LFSs, parts of them can be assigned to cloud servers to guarantee low dropping. However, delay will be higher than when the traffic is processed in LFSs.

Conclusions

In this article, the challenges and opportunities of applying fog computing in IoV are discussed, and a regional CFC-IoV architecture is proposed. CFC-IoV adopts a coordinator to provide low-delay coordination services for IoV applications. We emphasize the cooperation in fog computing, and the following four functions are discussed: mobility control, multi-source data acquisition, distributed computation and storage, and multi-path data transmission. To optimize the performance of CFC-IoV, a hierarchical resource management model is proposed, which includes intra-fog energy-aware resource management and inter-fog QoS-aware resource management. Essential simulation results are provided to prove the effectiveness of the proposed model.

References

[1] S. Djahel *et al.*, "A Communications-Oriented Perspective on Traffic Management Systems for Smart Cities: Challenges and Innovative Approaches," *IEEE Commun. Surveys & Tutorials*, 2015, vol. 17, no. 1, pp. 125–51.

[2] N. Fernando, S. W. Loke, and W. Rahayu, "Mobile Cloud Computing: A Survey," *Future Generation Computer Systems*, 2013, vol. 29, no. 1, pp. 84-106.

[3] A. Botta *et al.*, "On the Integration of Cloud Computing and Internet of Things.," *2014 Int'l. Conf. Future Internet of Things and Cloud*, 2014, Aug., pp. 23–30.

[4] J. Wan *et al*, "VCMIA: A Novel Architecture for Integrating Vehicular Cyber-Physical Systems and Mobile Cloud Computing," *Mobile Networks and Applications*, 2014, vol. 19, no. 2, pp. 153–60.

[5] H. Yao *et al.*, "Migrate or Not? Exploring Virtual Machine Migration in Roadside Cloudlet-Based Vehicular Cloud," *Concurrency and Computation: Practice and Experience*, 2015, vol. 27, no. 18, pp. 5780–92.

[6] L. M. Vaquero and L. Rodero-Merino, "Finding Your Way in the Fog: Towards a Comprehensive Definition of Fog Computing," *ACM SIGCOMM Computer Commun. Review*, 2014, vol. 44, no. 5, pp. 27–32.

[7] R. Yu *et al.*, "Toward Cloud-Based Vehicular Networks with Efficient Resource Management," *IEEE Network*, 2013, vol. 27, no. 5, pp. 48–55.

[8] X. Hou *et al*, "Vehicular Fog Computing: A Viewpoint of Vehicles as the Infrastructures," *IEEE Trans. Vehic. Tech.*, 2016, vol. 65, no. 6, pp. 3860–73.

[9] N. Truong *et al.*, "Software Defined Networking-Based Vehicular Adhoc Network with Fog Computing," *2015 IFIP/IEEE Int'l. Symp. Integrated Network Management*, May 2015, pp. 1202–07.

[10] X. Wang *et al.*, "Improved Rule Installation for Real-Time Query Service in Software-Defined Internet of Vehicles," *IEEE Trans. Intelligent Transportation Systems*, 2017, vol. 18, no. 2, pp. 225–35.

[11] C. Campolo *et al.*, "Modeling Broadcasting in IEEE 802.11 p/WAVE Vehicular Networks," *IEEE Commun. Letters*, 2011, vol. 15, no. 2, pp. 199–201.

[12] J. Petit *et al.*, "Pseudonym Schemes in Vehicular Networks: A Survey," *IEEE Commun. Surveys & Tutorials*, 2015, vol. 17, no. 1, pp. 228–55.

[13] M. Portnoy, *Virtualization Essentials*, vol. 19, Wiley, 2012.

[14] M. Shojafar *et al.*, "Energy-Saving Adaptive Computing and Traffic Engineering for Real-Time-Service Data Centers," *2015 IEEE ICC Wksp.*, June 2015, pp. 1800–06.

Biographies

WENYU ZHANG (wenyuzhang@bjtu.edu.cn) received his B.S. and M.S. degrees in communication engineering from Beijing Jiaotong University (BJTU), China, in 2013 and 2016, respectively, where he is currently working toward a Ph.D. degree in communication engineering. His research interests include protocols and resource management in fog computing and vehicle networks.

ZHENJIANG ZHANG (zhjzhang1@bjtu.edu.cn) received his Ph.D. degree in communication and information systems from BJTU in 2008. He became a professor at the same university in 2014. He is currently serving as the vice dean of the School of Software Engineering, BJTU. He has published about 70 professional research papers. His research interests include fog computing, communication protocols, and wireless sensor networks.

HAN-CHIEH CHAO [SM] (hcc@mail.ndhu.edu.tw) received his M.S. and Ph.D. degrees in electrical engineering from Purdue University, West Lafayette, Indiana, in 1989 and 1993, respectively. He is currently a joint appointed Distinguished Professor of National Ilan University, Taiwan. He has published about 400 refereed professional research papers. He is a Fellow of IET. His research interests include high-speed networks and wireless networks.