2016 IEEE 3rd International Conference on Cyber Security and Cloud Computing

Scalable Fog Computing with Service Offloading in Bus Networks

Dongdong Ye, Maoqiang Wu
Guangdong University
of Technology, China
Email: dongdongye8@163.com,
maoqiang.wu@vip.163.com

Shensheng Tang
Missouri Western State University, USA
Email:stang@missouriwestern.edu

Rong Yu
Guangdong University
of Technology, China
Email:yurong@ieee.org

***Abstract*—With the rapid increase of mobile devices, the computing load of roadside cloudlets is fast growing. When the computation tasks of the roadside cloudlet reach the limit, the overload may generate heat radiation problem and unacceptable delay to mobile users. In this paper, we leverage the characteristics of buses and propose a scalable fog computing paradigm with servicing offloading in bus networks. The bus fog servers not only provide fog computing services for the mobile users on bus, but also are motivated to accomplish the computation tasks offloaded by roadside cloudlets. By this way, the computing capability of roadside cloudlets is significantly extended. We consider an allocation strategy using genetic algorithm (GA). With this strategy, the roadside cloudlets spend the least cost to offload their computation tasks. Meanwhile, the user experience of mobile users are maintained. The simulations validate the advantage of the propose scheme.**

***Index Terms*—scalable, cloud computing, fog computing, bus networks, offloading.**

I. Introduction

With the rapid development of mobile communication technologies, the number of mobile devices is significantly increasing. Global mobile devices and connections in 2015 has grown to 7.4 billion [1]. The growth of mobile devices brings tremendous demand on executing rich media and data analysis applications, such as mailing, GPS routing, Internet banking, gaming, etc. Due to their own limited capabilities, mobile devices may have to offload their computation tasks to the available cloud [2]. However, the tremendous data size brings the large latency of transmission to the cloud [3].

To reduce the WAN latency, the concept of cloudlet is introduced as a trusted, resource-rich computer or cluster of computers well-connected to the Internet and being closer to the mobile devices [4]. For efficient cloud and cloudlet management, we proposed a hierarchial cloud architecture [5]. The architecture consists of three components, in which the cloudlets offer cloud services to nearby mobile users while the central cloud is used for global decisions. When the capability of the cloudlet reach the limit, the overload may generate heat radiation problem to the cloudlet and unacceptable delay to the mobile users. In order to solve this situation, fog computing is proposed as the extension of the cloudlets [6], [7]. On the one hand, fog servers provide computing service to mobile users as light-weight cloudlet servers. On the other hand, fog servers connect cloud to extend the computing capability.

However, few work employs bus fog servers to share the computation tasks offloaded by roadside cloudlets. Buses have advantages than other vehicles as computation offloading providers. In a bus network, the buses could deploy fog computing servers and gateways to provide service for the in bus mobile users [7]–[9]. The fixed mobility trajectories and strong periodicity are desirable for roadside cloudlets to assign the computation tasks. Furthermore, buses are public facilities as same as roadside cloudlets. Mobile users need not worry about privacy exposure problem.

In this paper, we consider a scalable fog computing paradigm with service offloading in bus networks. Bus fog computing servers are treated as the extension of the roadside cloudlets. The overload roadside cloudlets can offload part of their computation tasks to bus fog servers on bus. The computation capability of roadside cloudlets serving for mobile users can be improved in this paradigm. We consider an allocation strategy for roadside cloudlets to assign their computation tasks to bus fog servers. We find an optimal allocation solution using genetic algorithm. This solution enables the roadside cloudlets to spend the least cost and satisfy user experience.

The main contributions of this paper are summarized as follows.

- We analyze that the bus networks have unique characteristics: fixed mobility trajectories, strong periodicity, uniform fog servers and public facilities. These characteristics make buses better than other vehicles to share computation tasks from the overload roadside cloudlets.
- We consider a scalable fog computing paradigm with service offloading in bus networks. In the paradigm, the roadside cloudlets allow the bus fog servers to extend their computing capability serving for the mobile users within their communication coverage.
- We elaborately design an allocation strategy using genetic algorithm (GA). By the algorithm, the roadside cloudlets spend the least cost to offload their computation tasks to the bus fog servers. Meanwhile, the user experience of mobile users is maintained.

The rest of this paper is organized as follows. In Section II, we describe the scalable fog computing paradigm with service offloading in bus networks. We study the allocation strategy of

978-1-5090-0946-6/16 $31.00 © 2016 IEEE
DOI 10.1109/CSCloud.2016.34

the roadside cloudlet and use a genetic algorithm in Section III. Simulation results are presented in Section IV. We conclude the paper in Section V.

II. System Model

We consider a scalable fog computing paradigm with service offloading in bus networks as shown in Fig. 1. The paradigm includes two parts: roadside cloudlets and bus fog servers. The roadside cloudlet consists of three components: dedicated local servers, location-based service (LBS) providers, access points (APs). The dedicated local servers virtualize physical resource and act as a potential cloud computing site. The LBS providers offer the real time location of each bus in bus networks. The APs act as gateways for mobile users and bus fog servers within the communication coverage to access the roadside cloudlet. Mobile users have computationally intensive and delay sensitive tasks to be completed. They access APs and use the computing service of the roadside cloudlet. However, the service capacity of the roadside cloudlet is limited. When the roadside cloudlet exceeds the limit, The overload roadside cloudlet will bring the mobile users unacceptable user experience.

The bus fog server is a virtualized computing system on bus, which is similar to a light-weight cloudlet server. They not only provide computing, storage and wireless communication to the mobile users on bus, but also afford avaiable computing resource for computation tasks offloaded by the roadside cloudlet. In the paradigm, the bus fog servers are the extension of the roadside cloudlet. The roadside cloudlet can extend different scale of computing capability serving for the mobile users by allowing different number of bus fog servers.

The scalable fog computing paradigm works as a cyclical mechanism. The roadside cloudlet makes an estimation for the total volume of the computation tasks exceeding the limit every cycle. The LBS providers afford the realtime movement situation of the buses within the communication coverage of the roadside cloudlet. The roadside cloudlet makes a strategy of allocation the computation tasks. Finally this offloads the computation tasks to the bus fog servers via APs and offers them incentive payment according to the allocation strategy. The allocation strategy not only minimizes the cost of transmitting data and incentive to bus fog servers, but also satisfies the user experience of mobile users.

III. Problem Formulation and Solution

In this section, we study the incentive strategy of the roadside cloudlet and use a genetic algorithm to solve it.

A. Problem Formulation

The scalable fog computing paradigm works as a cyclical mechanism. In one cycle there are a set of mobile users N whose computation tasks need to be offloaded by the roadside cloudlet to bus fog servers. The computation task of the mobile user $n \in N$ is denoted as $S_n \triangleq (C_n, D_n, T_n)$. Here C_n denotes the total number of CPU cycles required to accomplish the commutation task S_n. D_n denotes the data size involving in

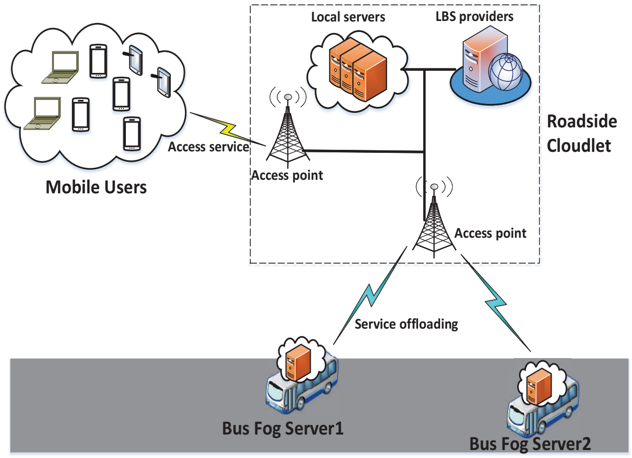

Fig. 1: Scalable fog computing paradigm with service offloading in bus networks.

the computation task S_n. T_n denotes the tolerance time of the mobile user n for completing the computation task S_n. T_n is related with the user experience of the mobile user n .

The coordinate of the AP is (x_o, y_o). The scalable fog computing paradigm includes a set of buses M within the communication coverage of the AP in one cycle. Let $l_m = \{(x_1, y_1), (x_2, y_2), ...(x_a, y_a)\}$ be the two-dimensional coordinate trajectory of the bus $m \in M$. The distance between the AP and the moving trajectory l_m can be obtain by

$$d_m = (x_m - x_o)^2 + (l(x_m) - y_o)^2, d_m \leq r, \quad (1)$$

where r is the maximum communication radius of the AP. The fog server on the bus m is denoted as $B_m \triangleq (F_m, R_m, Q_m)$. Here F_m denotes the computation capability (CPU cycles per second) of the bus fog server B_m. Q_m denotes the storage capability of the bus fog server B_m. R_m denotes the transmission rate from the roadside cloudlet to the bus fog server B_m. From [10], the transmission rate R_m can be obtain by

$$R_m = W log_2(1 + \frac{P_r {d_m}^{-\eta} |h_0|^2}{N_0}). \quad (2)$$

Here W is the channel bandwidth. P_r is the transmit power of the AP. h_0 is the complex Gaussian channel coefficient that follows the complex normal distribution $CN(0, 1)$. N_0 is the additive white Gaussian noise (AWGN) at the bus receivers. To maintain stable transmission, the rate of transmission must keep ${R_m}^*$. The transmit power of the AP can be obtain by

$$P_r = \frac{(2^{{R_m}^*/W} - 1)N_0}{d_m |h_0|^2}. \quad (3)$$

Since W, h_0, N_0 are constants, P_r is proportional to d_m. The AP needs to increase its power to keep the rate of transmission, when the bus m is leaving away. Denote $|l_m|$ as the moving trajectory of the bus m within the communication coverage of the AP. The total time of the bus m within the communication coverage of the AP is denoted as $T_m = \frac{|l_m|}{v}$, where v is the average speed of the bus m. The average power of the AP transmitting to the bus m is denoted as

$$\overline{P_r} = \frac{\int_{|l_m|} P_r}{|l_m|}. \quad (4)$$

The time of the AP transmitting the computation task S_n to the bus m can be obtain by $\frac{D_n}{R_m}$. The time of the bus m that accomplishes the computation task S_n can be obtain by $\frac{C_n}{F_m}$. The time of the bus B_m transmitting the computing result back to the roadside cloudlet is negligible because of the insignificant data size of computing result [11]. Therefore, the total time of the bus m service for the mobile user n is denoted as

$$t_n^m = \frac{D_n}{{R_m}^*} + \frac{C_n}{F_m}. \quad (5)$$

Before receiving the offloaded computation tasks from the cloudlet, the bus fog server B_m is serving for q_m computation tasks of mobile users in bus. Let X_{nm} be binary indicator. $X_{nm} = 1$ if D_n is offloaded to B_m and otherwise $X_{nm} = 0$. We denote the willingness of B_m for receiving a set of offloaded computation tasks $X_{nm}D_n$ as $w_m{\cdot}\log_2(1 + c_r{\cdot}\sum_{n\in K_m} X_{nm}D_n)$, where $w_m = (Q_m - q_m)/Q_m$ is the factor and c_r is the unit data price offered by the roadside cloudlet. The bus B_m has higher willingness to receive the offloaded computation tasks with less q_n. From [11], we know that the energy consumption for computation tasks is inversely proportional to the computation capability ${F_m}^2$. The bus fog servers consume higher energy for accomplishing the computation tasks with higher CPU cycles per second. In this paper, we take the computing cost as $10^{-11}{F_m}^2\sum_{n\in K_m} X_{nm}D_n$, K_m denotes the number of users serviced by the bus m. The Utility of the bus fog server is denoted as

$$U_m = w_m{\cdot}\log_2(1{+}c_r{\cdot}\sum_{n\in K_m} X_{nm}D_n){-}10^{-11}{F_m}^2\sum_{n\in K_m} X_{nm}D_n. \quad (6)$$

Clearly, the utility function of the bus fog server is concave, which indicates that the maximum value of this function exists. Therefore, given first order optimality condition, we obtain

$$c_r = \frac{10^{-11}{F_m}^2}{w_m - 10^{-11}{F_m}^2\sum_{n\in K_m} X_{nm}D_n}. \quad (7)$$

The bus fog server is willing to receive the offloaded computation tasks only if its maximum utility is satisfied. The roadside cloudlet needs to afford higher c_r if it offload more computation tasks to B_m.

The incentive cost that the roadside cloudlet pays for the bus fog server can be obtain by $r_n{\cdot}\sum_{n\in K_m} X_{nm}D_n$. The transmission cost of the roadside cloudlet can by given by $\beta\overline{P_r}\frac{\sum_{n\in K_m} X_{nm}D_n}{{R_r}^*}$, where β is a factor and $\overline{P_r}\frac{\sum_{n\in K_m} X_{nm}D_n}{{c_m}^*}$ is the total energy consumption of transmitting computation tasks. Therefore, the total cost of the roadside cloudlet is denoted as

$$I = \sum_{m\in M}(c_n{\cdot}\sum_{n\in K_m} X_{nm}D_n + \beta\overline{P_r}\frac{\sum_{n\in K_m} X_{nm}D_n}{{R_r}^*}). \quad (8)$$

Our problem is to find an optimal allocation strategy for the overload roadside cloudlet to offload the computation tasks to the bus fog servers. It can be obtain by:

$$MinI \quad (9)$$

$$s.t.maxU_m \quad (10)$$

$$t_n^m{\le}T_n{\le}T_m \quad (11)$$

$$\sum_{n\in K_m} X_{nm}D_n + q_m{\le}Q_m \quad (12)$$

$$\sum_{m=1}^{M} X_{nm} = 1. \quad (13)$$

The Eqn. (10) satisfies the maximum utility of each bus fog server. The Eqn. (11) ensures that the time of accomplishing computation tasks satisfies the user experience. The Eqn. (12) restricts that the total data size of computation tasks is less than the storage capability of the bus fog server. The Eqn. (13) ensures each computation task is only associate to a bus fog server. This problem is an NP-hard problem. We use a genetic algorithm (GA) to solve the problem.

B. Solution:GA

To solve the NP-hard problem, we suggest a GA. By the algorithm, the system can find out the near-optimal budget R^* to maximize the utilities of buses .The steps of the strategy are given as follows:

- **Step 1:** Initialization. Input the computation tasks N and label them with different number j, $1{\leqslant}j{\leqslant}N$. Set the maximum number of iterations X and the population size Z.
- **Step 2:** Sort the label of the computation tasks randomly as one solution and get an initial population of Z solutions. To each solution, set $M - 1$ breakpoints. The M fragments of computation tasks are assigned to the M bus fog servers. When a solution meets the subject (10) (11) (12) (13), it is an effective solution. To each effective solution, the roadside cloudlet adjusts offering unit price r_n of each computation task to make U_m maximum. Then it calculates the total cost I by Eqn. (8) as the fitness.
- **Step 3:** Record the solution with the highest as the local optimal solution of the population. Select numbers of the solutions with the highest fitness as the first generation subpopulation of solutions. Generate a second generation subpopulation of solutions from the first generation subpopulation through a combination of genetic operators: crossover (also called recombination), and mutation.
- **Step 4:** Produce the solutions randomly and add the second generation subpopulation of solutions to a new population of X solutions. The the number of iterations is added by one.

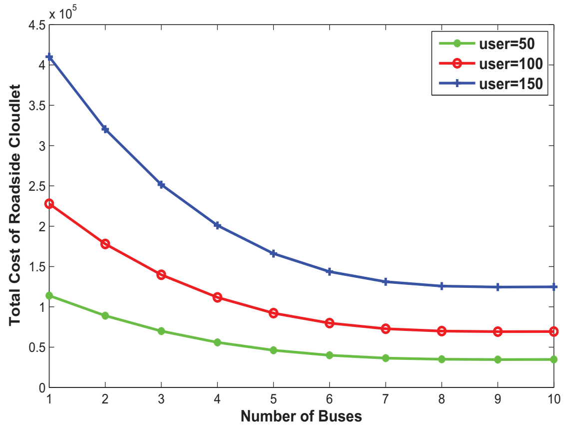

Fig. 2: Total cost of service offloading.

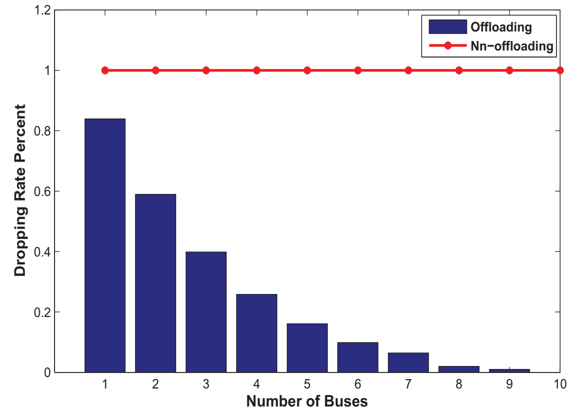

Fig. 3: Performance comparison between service offloading and non-offloading.

- **Step 5:** If the number of iterations does not meet X, jump to Step 2, else compare of the fitness of the X local optimal solution and select the solution with the highest fitness as a near-optimal solution.
- **Step 6:** Output the near-optimal solution.

IV. Numerical Results

In this section, we evaluate the performance of the proposed incentive strategy by simulations. We use the traffic trace data set of the city buses [12]. The data set collected traces of the actual movement of the fleet of city buses in the Seattle, Washington metropolitan area, on their normal routes providing passenger bus service throughout the city. We randomly select an area of $2km \times 2km$ in the map of the data set. The data set reflects the actual movement of the buses along with the time. Therefore the number and the mobility trajectories of the buses in the cycle time depend solely on the actual movement data. The roadside cloudlet is assumed to be located at the center of the area. We use the following parameter settings as that in [3], [11]: $F_m = 50 \backsim 100GZ$, $W = 6Mpbs$, $R_m = 2 \backsim 3Mb/s$, $C_n = 200 \backsim 1000Mcycles$, $D_n = 200 \backsim 1000KB$.

A. Total cost of computation offloading

Fig. 2 shows the result for the total cost of the roadside cloudlet over the number of the buses $|M|$ for three values of the number of mobile users $|N|$. For fixed $|N|$, the total cost I of the roadside cloudlet declines with the increase of $|M|$. It is because that the increasing number of bus fog servers brings more options for the roadside cloudlet to offload the computation tasks. The bus fog servers are willing to accomplish the offloaded computation tasks with low incentive. Therefore, the roadside cloudlet spends lower cost for offloading. With more mobile users, the roadside cloudlet spends more cost. It is because that the bus fog servers need more incentive to satisfy their optimal utilities when the number of mobile users increases. Therefore, the roadside cloudlet has to increase the unit price to motivate the bus fog servers to receive the offload computation tasks.

B. Performance comparison between computation offloading and non-offloading

Fig. 2 shows the result for the dropping rate of mobile users over the number of the buses $|M|$ with $|N| = 100$. The dropping rate denotes the percentage of the computation tasks that can not satisfy the user experience of mobile users. With higher dropping rate, the more mobile users have low user experience. The line chart shows that finishing time of all the computation tasks do not less than the the tolerance time of mobile users. The bar chart shows that the dropping rate inclines with the increase of the $|M|$. It is because that the computation capability grows with more bus fog servers.

V. Conclusion

In this paper, we considered a scalable fog computing paradigm in bus networks. The bus fog server is the extension of the roadside cloudlets. The roadside cloudlets extend computation capability from the bus fog servers in the paradigm. We considered an allocation strategy of the roadside cloudlet to offload the computation tasks to bus fog servers. We used a genetic algorithm to find the optimal allocation solution. The simulation results showed that the roadside cloudlet spends the least cost while all the mobile users have the better user experience via our strategy.

VI. Acknowledge

Rong Yu is the corresponding author of this paper. The work is supported in part by programs of NSFC under Grant nos. 61422201, 61370159 and U1201253, U1301255, the Science and Technology Program of Guangdong Province under Grant no. 2015B010129001, Special-Support Project of Guangdong Province under grant no. 2014TQ01X100, High Education Excellent Young Teacher Program of Guangdong Province under grant no. YQ2013057, Science and Technology Program of Guangzhou under grant no. 2014J2200097 (Zhujiang New Star Program).

REFERENCES

[1] C. V. N. I. Cisco, "Global mobile data traffic forecast update, 2014–2019," *white paper*, 2015.

[2] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: architecture, applications, and approaches," *Wireless communications and mobile computing*, vol. 13, no. 18, pp. 1587–1611, 2013.

[3] H. Zhang, Q. Zhang, and X. Du, "Toward vehicle-assisted cloud computing for smartphones," *Vehicular Technology, IEEE Transactions on*, vol. 64, no. 12, pp. 5610–5618, 2015.

[4] T. Verbelen, P. Simoens, F. De Turck, and B. Dhoedt, "Cloudlets: bringing the cloud to the mobile user," in *Proceedings of the third ACM workshop on Mobile cloud computing and services*, pp. 29–36, ACM, 2012.

[5] R. Yu, Y. Zhang, S. Gjessing, W. Xia, and K. Yang, "Toward cloud-based vehicular networks with efficient resource management," *Network, IEEE*, vol. 27, no. 5, pp. 48–55, 2013.

[6] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, pp. 13–16, ACM, 2012.

[7] T. H. Luan, L. Gao, Z. Li, Y. Xiang, and L. Sun, "Fog computing: Focusing on mobile users at the edge," *arXiv preprint arXiv:1502.01815*, 2015.

[8] R. Hussain, F. Abbas, J. Son, S. Kim, and H. Oh, "Using public buses as mobile gateways in vehicular clouds," in *Consumer Electronics (ICCE), 2014 IEEE International Conference on*, pp. 175–176, IEEE, 2014.

[9] B. Brik, N. Lagraa, A. Lakas, and Y. Ghamri-Doudane, "Finding a public bus to rent out services in vehicular clouds," in *Vehicular Technology Conference (VTC Fall), 2015 IEEE 82nd*, pp. 1–7, IEEE, 2015.

[10] Y. Zhang, E. Pan, L. Song, W. Saad, Z. Dawy, and Z. Han, "Social network aware device-to-device communication in wireless networks," *Wireless Communications, IEEE Transactions on*, vol. 14, no. 1, pp. 177–190, 2015.

[11] X. Chen, "Decentralized computation offloading game for mobile cloud computing," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 26, no. 4, pp. 974–983, 2015.

[12] J. G. Jetcheva, Y.-C. Hu, S. PalChaudhuri, A. K. Saha, and D. B. Johnson, "CRAWDAD dataset rice/ad_hoc_city (v. 2003-09-11)." Downloaded from http://crawdad.org/rice/ad_hoc_city/20030911, sep 2003.