# Virtual Machine Placement for Backhaul Traffic Minimization in Fog Radio Access Networks

Ya-Ju Yu[1], Te-Chuan Chiu[2], Ai-Chun Pang[2,3,4], Ming-Fan Chen[3], and Jiajia Liu[5]
[1]Department of Computer Science and Information Engineering, National University of Kaohsiung, Kaohsiung, Taiwan
[2]Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan
[3]Graduate Institute of Networking and Multimedia, National Taiwan University, Taipei, Taiwan
[4]Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan
[5]School of Cyber Engineering, Xidian University, Xi'an, China
E-mail: yjyu@nuk.edu.tw, {d01922009, acpang, r03944059}@csie.ntu.edu.tw, liujiajia@xidian.edu.cn

***Abstract*—With the advance of wireless technologies, the rapid mobile data traffic growth will lead to severe network resource consumption and exceptionally long latency to access services, especially for cloud-based applications. To tackle these issues, Fog Radio Access Network (F-RAN) is recently emerged for next generation cellular networks. F-RAN is considered as an extension of the cloud computing paradigm to the edge of the network and a highly virtualized platform that provides computing, storage, and network services for mobile devices. However, how to appropriately place virtual machines (VMs) into fog nodes in F-RAN systems is very challenging, and will significantly affect the bandwidth consumption of backhaul links. Thus this paper studies the replication-based VM placement problem, and aims at minimizing the total backhaul traffic generated by VM migrations and data transmissions. We observe that the VM placement operation should not be frequently executed and practically considers the problem in a long term aspect. Then we propose a heuristic algorithm to solve the problem. The simulation results agree our observation and show that compared with a greedy approach and an optimal algorithm, the proposed algorithm demonstrates with favorable results for the overall backhaul network usage.**

***Index Terms*—Cellular networks, Fog radio access networks, VM replication**

## I. Introduction

According to the latest report announced by Cisco [1], global mobile data traffic is increasing at an astounding pace. There is no doubt that a great amount of the mobile data traffic will be contributed by real-time interactive services (such as virtual reality and augmented reality) and mission-critical Internet-of-Things (IoT) applications. That is why reliable and low latency communications have been considered as one of the critical issues by standardization bodies for future mobile system design. Although cloud data centers provide a centralized abundant resource pool for mobile applications, the rapid data traffic growth leads to severe bandwidth consumption [2], and end users suffer long service latency when their services are provided at cloud data centers. To tackle those issues, *Fog Radio Access Networks (F-RAN)* [3], an evolved and notable concept for next generation cellular networks, are emerged recently. F-RAN is considered as an extension of the cloud computing paradigm to the edge of the network and a highly virtualized platform that provides computing, storage, and network services for mobile devices [4].

An F-RAN consists of several fog nodes, and each fog node resides in a base station or a small/femto access point. With virtualization technologies, a fog node has the capability to run multiple virtual machines (VMs) on its own physical machine simultaneously, and a VM can be duplicated into multiple copies and placed in multiple fog nodes. The VMs can be flexibly placed in F-RAN [5], [6], based on the traffic distribution and moving pattern of mobile users. However, the dynamic VM placement in F-RAN systems incurs a significant cost on bandwidth consumption of backhaul network links. The reason is described as follows. When a mobile user is in the coverage area of a fog node which has the application VM requested by the user, the service can be provided by the fog node without consuming any backhaul bandwidth. On the other hand, if the fog node does not host the application VM, to serve the user, extra backhaul data traffic will be generated by one of the following two cases. (1) The corresponding VM will be copied from some fog node owning the VM to the fog node in which the user resides through the backhaul network. In this case, the generated traffic is termed “VM traffic”, and the amount of consumed backhaul bandwidth is related to the size of the application VM. (2) The user can directly access the application via the backhaul network to another fog node which has the VM for the application requested by the user. In this situation, the bandwidth of backhaul network is consumed by the user's access for the application service, and the generated traffic is termed “data transmission traffic”.

We can observe that frequent VM re-placement will cause serious “VM traffic” while static VM placement may result in significant “data transmission traffic” because fixed VM location cannot adapt to the property of user mobility. Backhaul links are considered as the bottleneck of radio access networks and are becoming more congested as advanced wireless technologies, like coordinated multi-point transmissions, are adopted. Therefore, the VM placement problem for F-RAN in minimizing the backhaul bandwidth has to consider several issues: (1) to determine how many replicas each VM should have and in which fog nodes the replicas should be placed; (2) to decide which VM replica each user will be associated to for accessing application services. Furthermore, to reflect the practical situations for VM replication overheads

978-1-4673-8999-0/17/$31.00 ©2017 IEEE

in hardware and software, the VM placement cannot be frequently executed.

There have been numerous researches studying the VM placement problem in cloud data centers. One of the objectives is to minimize the communication traffic between virtual machines [7], [8]. However, in cloud data centers, the VM placement problem does not consider user location which have a significant impact in F-RAN. Recently, a few researches have investigated the VM placement problem to minimize the usage of backhaul links in wireless networks with the consideration of user mobility [9], [10]. Specifically, in these works, the VM placement problem is formulated under the assumption that each VM has only one copy and serves one user. Furthermore, the tradeoff between the VM traffic and the data transmission traffic would not be well dealt with since these works do not consider the target scenario in a long term aspect.

In this paper, we study the replication-based VM placement problem for F-RAN systems. The objective is to minimize the amount of traffic generated by the VM traffic and data transmission traffic. The contributions of this paper are elaborated as follows. 1) This paper is one of the first works studying the replication-based VM placement problem in F-RAN and practically considers the problem in a long term aspect. We observe that VM replacement should not be frequently executed to actually minimize the total traffic generated in backhaul network. 2) We propose a heuristic algorithm with the consideration of the long-term decision period to tackle the problem. 3) Compared with a greedy method and an optimal solution, the proposed algorithm demonstrates with favorable results for the overall backhaul network usage. The simulation results also justify our observation and show some useful insights into the replication-based VM placement problem.

The remainder of this paper is organized as follows. Section II presents the system model and the formal formulation of the target problem. In Section III, the proposed algorithm is demonstrated, and the time complexity is also evaluated. Performance evaluation is shown in Section IV. Finally, Section V concludes this work.

## II. Sysyem Model and Problem Formulation

### A. System Model

In next-generation cellular systems, fog radio access networks (F-RAN) with virtualization technologies are one of promising technologies to reduce the backhaul network consumption. How to place VMs with replication mechanism in each fog node is an important problem, referred to *replication-based VM placement problem*. Since each fog node has its resource (e.g., computing and storage) limitations, the number of VMs that can be activated in a fog node is limited. When a fog node operates an application VM, the fog node should consume CPU and storage resources. In this paper, a VM is associated with an application and can serve the limited number of users who request the same application. In other words, the same application may have multiple VMs in order to serve the users or to reduce the backhaul traffic.

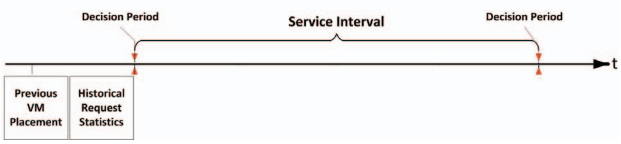

Fig. 1. Time Slotted Model of F-RAN

When a user connects to a fog node and requests an application service, if the fog node already has the application VM, the user can enjoy his/her service without consuming the backhaul bandwidth. Otherwise, we have two ways to support the application service for the user by consuming backhaul bandwidth. 1) We can move or duplicate the application VM from another fog node to the anchor fog node connecting with the user. This backhaul traffic is called *VM traffic*. 2) On the other hand, we do not replicate the application VM and the application service is provided by a remote fog node with the application VM. The application data is transmitted from the remote fog node to the anchor fog node for the user. This backhaul traffic is called *data transmission traffic*. We can observe that frequent VM re-placement will cause serious VM traffic while static VM placement may result in significant data transmission traffic. There is a tradeoff between the VM traffic and the data transmission traffic. In order to minimize the backhaul traffic, we practically consider that the VM placement decision can be made in a long term period based on the statistics of historical user activities shown in Fig. 1. Specifically, during the service interval, the VM placement will not be changed again. When we consider the long-term VM placement, the placement problem will be more complicated because we have to ensure that the total data traffic cannot exceed a single backhaul link under the long-term VM placement decision.

### B. Problem Formulation

In this paper, we study the replication-based VM placement problem in F-RAN. The objective is to minimize the overall backhaul network traffic during a long term period.

We consider a network graph $G = (V, E)$ with $N$ users and a service interval $T$, where there are multiple time slots in the service interval. Each fog node $i$, $i \in V$, has computing capacity $c_i$ and storage capacity $s_i$. An edge $(u, v) \in E$ with a bandwidth $b(u, v)$ represents a backhaul link between fog node $u$ and fog node $v$. If application $k$, $k \in A$, running on the corresponding VM, a fog node should consume computing resource $c^k$ and storage resource $s^k$, where $A$ is the set of applications. When a user would like to access application $k$, the network bandwidth demand is $b^k$. Specifically, when a user has to access the required application from a remote fog node, the data transmission traffic $b^k$ will be generated in the edges of the path. Moreover, the maximum service requests that can be served by a VM for application $k$ is $r^k_{\max}$. Because of the computing and storage limitations, the number of VMs in each fog node is limited as well. Intuitively, the VMs of

TABLE I
SUMMARY OF NOTATIONS

| Symbol | Description |
|---|---|
| $T$ | The service time interval |
| $G$ | The topology of F-RAN system |
| $V$ | The set of fog nodes |
| $E$ | The set of backhaul edges |
| $c_i$ | The computing capacity of fog node $i$ |
| $s_i$ | The storage capacity of fog node $i$ |
| $b(u,v)$ | The bandwidth of edge $(u,v)$ |
| $N$ | The number of users |
| $A$ | The set of applications |
| $c^k$ | The computing requirement of application $k$ |
| $s^k$ | The storage requirement of application $k$ |
| $b^k$ | The network bandwidth demand of application $k$ |
| $r^k_{\max}$ | The maximum number of requests for application $k$ that can be serviced by a VM |
| $R$ | The historical request statistics, where $r^{(t)}_{i,k} \in R$ |
| $r^{(t)}_{i,k}$ | The number of requests of application $k$ from fog node $i$ at time slot $t$ |
| $I^{pre}_{i,k}$ | An indicator function, which is 1 if fog node $i$ served application $k$ at previous service interval, and 0 otherwise |
| $I_{i,k}$ | An indicator function, which is 1 if fog node $i$ serves application $k$ currently, and 0 otherwise |
| $I^{REPL}_{i\to j,k}$ | An indicator function, which is 1 if application $k$ duplicated from fog node $i$ to fog node $j$, and 0 otherwise |
| $\delta^{TRAN}_{i\to j,k}$ | The fog node $j$'s service ratio of arrived requests to total requests of application $k$ from fog node $i$ |
| $f^{REPL}_{i\to j,k}(u,v)$ | The VM traffic of application $k$ from fog node $i$ to fog node $j$ through edge $(u,v)$ |
| $f^{(t)TRAN}_{i\to j,k}(u,v)$ | The total data transmission traffic of application $k$ from fog node $i$ to fog node $j$ through edge $(u,v)$ at time slot $t$ |
| $D^{REPL}$ | The total VM traffic in total service interval |
| $D^{TRAN}$ | The total data transmission traffic in total service interval |

each application $k$ may be more than one in order to serve users more than $r^k_{\max}$.

Since each user will move and send different application requests in the service time interval, we statistically collect the historical activities of users as a request statistics $R$. $r^{(t)}_{i,k}$ records the number of requests of application $k$ from fog node $i$ at time slot $t$, where $r^{(t)}_{i,k} \in R$. Under the previous VM placement result, $I^{pre}_{i,k}$ is an indicator function, which is 1 if fog node $i$ served application $k$ at previous service interval, and 0 otherwise. Given the previous VM placement result, when a fog node does not have the VM serving the application requested by some users, we should determine the application VM should be replicated from another fog node or not. If the VM of application $k$ is replicated from fog node $j$ to fog node $i$, indicator function $I^{REPL}_{i\to j,k}$ is 1 and the VM traffic will be generated. The VM traffic of application $k$ from fog node $i$ to fog node $j$ through edge $(u,v)$ is represented as $f^{REPL}_{i\to j,k}(u,v)$. The VM traffic for application $k$ on edge $(u,v)$ is $I^{REPL}_{i\to j,k} \times f^{REPL}_{i\to j,k}(u,v)$. Note that the VM traffic of application $k$ from fog node $i$ to fog node $j$ may pass through multiple edges.

When we determine that an application VM should not be replicated from another fog node, the data transmission traffic will be generated. At time slot $t$, the number of requests (users) for application $k$ from fog node $j$ is represented as $r^{(t)}_{j,k}$ and the network bandwidth requirement for accessing application $k$ is $b^k$. $\delta^{TRAN}_{i\to j,k}$ is used to represent the percentage of the user requests for application $k$ that should be served from fog node $i$ to fog node $j$. Therefore, the amount of the data transmission traffic for application $k$ from fog $i$ to fog $j$ on edge $(u,v)$ at time slot $t$ is $\delta^{TRAN}_{i\to j,k} \times r^{(t)}_{j,k} \times b^k$. Note that the data transmission traffic of application $k$ from fog node $i$ to fog node $j$ may pass through multiple edges. We define $f^{(t)TRAN}_{i\to j,k}(u,v)$ used to represent the amount of the data transmission traffic of application $k$ from fog node $i$ to fog node $j$ through edge $(u,v)$ at time slot $t$ as follows. Because we consider the VM placement decision in a long term period, the VM placement will consider the user distribution in each time $t$ to minimize the amount of the data transmission traffic, $\forall t \in T$, where $T$ is the service interval. A replication-based VM placement is feasible if the following constraints are met.

*Computing and storage constraints:* The computing and storage resource of each fog node $i$ for activating VMs cannot exceed the computation and storage capacities, where $I_{i,k}$ is an indicator function, which is 1 if fog node $i$ serves application $k$ in current service interval.

$$\sum_{k\in A} I_{i,k} \times c^k \le c_i, \forall i \in V \tag{1}$$

$$\sum_{k\in A} I_{i,k} \times s^k \le s_i, \forall i \in V \tag{2}$$

*User service constraint:* Each user application request has to be provided by at least one VM.

*Edge bandwidth constraint:* The summation of the VM replication flows and data transmission flows passing through an edge cannot exceed the edge network bandwidth at a single time slot $t$. Therefore, the VM placement may not be feasible due to the edge bandwidth constraint. This is why our problem is more complicated when we consider the long-term VM placement.

Now we define the target problem formally as follows.

**Replication-based VM Placement Problem**

*Input instance:* Consider the service time interval $T$, the network graph $G$ with the fog node set $V$ and the backhaul edge set $E$, the number of users $N$, the application set $A$, the request statistics $R$, and the previous service configuration of application $k$ in each fog node $i$ (i.e., $I^{pre}_{i,k}, \forall i \in V, k \in A$). Each fog node $i$ has computing resource $c_i$, storage resource $s_i$. Each application $k$ has computing resource requirement $c^k$, storage resource requirement $s^k$, and network bandwidth demand $d^k$.

*Objective:* The objective is to find a feasible VM placement configuration such that the overall backhaul network traffic in a long term period is minimized. The overall network traffic consists of the two main parts: the VM traffic and the data transmission traffic in Eq. (3) and (4), respectively.

$$D^{REPL} = \sum_{i\in V}\sum_{j\in V}\sum_{k\in A}\sum_{(u,v)\in E} I^{REPL}_{i\to j,k} \times f^{REPL}_{i\to j,k}(u,v) \tag{3}$$

$$D^{TRAN} = \sum_{t \in T} \sum_{i \in V} \sum_{j \in V} \sum_{k \in A} \sum_{(u,v) \in E} f_{i \to j,k}^{(t)TRAN}(u,v) \tag{4}$$

We state the objective function formally as

$$\min(D^{REPL} + D^{TRAN})$$

subject to *computing and storage constraints*, *user service constraint* and *edge bandwidth constraint* mentioned above. All symbols and variables are summarized in Table I.

## III. Replication-based Virtual Machine Placement

In this section, we prove that target problem is $\mathcal{NP}$-hard, and present an efficient algorithm to tackle replication-based virtual machine placement problem. Then, we analyze the time complexity of the proposed algorithm and show that it is a polynomial-time algorithm.

### A. Problem Hardness

**Theorem 1.** *The replication-based VM placement problem is $\mathcal{NP}$-hard.*

*Proof:* This problem obviously is $\mathcal{NP}$-hard and can be proved by a reduction from *two-commodity integral flow problem* problem [11]. The proof is omitted due to lack of space. ■

### B. Algorithm Description

Since our target problem is $\mathcal{NP}$-hard, we propose an efficient heuristic algorithm, named replication-based VM placement algorithm (RBP), which has two main procedures, *Priority Setting Mechanism* and *Candidate Searching Mechanism*, to solve the replication-based VM placement problem in polynomial time. For *Priority Setting Mechanism*, the goal is to decide the suitable request order with the consideration of critical flows. Besides, the service request sequence will divide into two groups to avoid violating the user service constraint. For *Candidate Searching Mechanism*, the procedure will search the most suitable candidates of fog nodes to place the replicas for each application. Since there exists a tradeoff between the VM traffic and the accumulation of the data transmission traffic, the proposed scheme will not only carefully deal with this non-trivial tradeoff but also consider the picking of victim application VMs to be deleted due to the fog resource limitations.

The proposed algorithm, as shown in Algorithm 1, starts with the initialization of parameters in Line 1. For all of service requests, we trigger *Priority Setting Mechanism()* to decide the order for requests according to their importance in Line 2. Next, in order to decide the placement for replicas of applications one after another, we execute *Candidate Searching Mechanism()* with the order of output from the priority setting mechanism in Line 3-4. The candidate searching mechanism will decide the best configuration with the minimum total network traffic for all requests and the output of the problem, including the configuration of each VM and each server ($I_i, I_{i,k}$), and also the backhaul network traffics of VM replication and of data transmission ($D^{REPL}, D^{TRAN}$) in Line 5-10. Finally, The output of the problem is decided after all requests accomplishing the candidate searching mechanism in Line 11.

**Algorithm 1** Replication-based VM Placement Algorithm

```
Input: $T, G, N, A, R, I_{i,k}^{pre}$
Output: $I_i, I_{i,k}$, $D^{REPL}, D^{TRAN}$
 1: $I_i, I_{i,k} \leftarrow 0, \forall i \in V, k \in A$, and $D^{REPL}, D^{TRAN} \leftarrow 0$
 2: Priority Setting Mechanism()
 3: for $r_{i,k} \in \hat{R}, r_{i,k} \in \check{R}$ and $r_{i,k} \neq 0$ do
 4:     Candidate Searching Mechanism()
 5:     $D^{REPL} \leftarrow D^{REPL} + \hat{D}^{REPL}$
 6:     $D^{TRAN} \leftarrow D^{TRAN}$+ Network Simplex($i, \widehat{sol}, r_{i,k}$)
 7:     if $\widehat{sol} \neq -1$ then
 8:         $I_{\widehat{sol}} \leftarrow 1$, and $I_{\widehat{sol},k} \leftarrow 1$
 9:     if $vic \neq -1$ then
10:         $I_{\widehat{sol},a} \leftarrow 0$
11: return $I_i, I_{i,k}$, $D^{REPL}, D^{TRAN}$
```

*1) Priority Setting Mechanism:* In the procedure 1, the main idea is to place the VM for serving the most critical flows first. Since the critical flows are those flows with higher bandwidth requirement for each request or with larger number of requests, it obviously causes more network usage compared with other flows passing the same number of hops. Besides, the VM replication scheme only executes during the decision period according to the time slotted model, it is more rational to consider previous sum of the requests from the historical request statistics as a useful reference when making the VM placement decision. Therefore, we will record the products of bandwidth demand for each request and the sum of previous requests in the historical request statistics for each application from each server (Line 4-6). Then, we will sort the products results in a descending order in Line 7 and 14. Finally, in order to prevent the replicas of applications for those critical flows occupying the limited resources and violating the user service constraint, the sorting sequence will split into two parts. In the first part, the scheme will place one VM for each concerned application as a guarantee, and place more replicas if the constraints are obeyed for the second part (Line 8-13).

*2) Candidate Searching Mechanism:* Then, the scheme will invoke the procedure 2 for serving requests of different applications from different servers in turn. The main idea is to search the nearby fog nodes which have the higher priority to receive the service requests with the closer distance to the local server by using breadth-first search from the local node (Line 3). The initialization is shown in Line 1-2, and $D^{pre}(i,k)$ represents as the data transmission traffic caused by the previous configuration, where $i$ is the source node and $k$ is the concerned application. However, to accomplish every request for deciding 1) using VM replication in the local server or 2) transmitting/receiving data traffic to/from the remote candidate server is a non-trivial problem, it faces two critical issues to tackle: one is the tradeoff between the VM traffic and the accumulation of data transmission traffic

```
Procedure 1 : Priority Setting Mechanism()
 1: $\hat{r}_{i,k}, \check{r}_{i,k} \leftarrow 0, \forall i \in V, k \in A, \hat{r}_{i,k} \in \hat{R}, \tilde{r}_{i,k} \in \check{R}$
 2: for $k \in A$ do
 3:     $c \leftarrow 0, \tilde{r}_i \leftarrow 0, \forall i \in V$, and $\bar{r}_i \in \tilde{R}$
 4:     for $t \in T$ do
 5:         for $i \in V$ do
 6:             $\tilde{r}_i \leftarrow \tilde{r}_i + r_{i,k}^{(t)} \times b^k$
 7:     Sort $\tilde{R}$ in a decreasing order
 8:     for $\tilde{r}_i \in \tilde{R}$ do
 9:         if $c < \left\lceil \frac{N}{r_{\max}^k} \right\rceil$ then
10:             $\hat{r}_{i,k} \leftarrow \tilde{r}_i$
11:         else
12:             $\check{r}_{i,k} \leftarrow \tilde{r}_i$
13:         $c \leftarrow c + 1$
14: Sort $\hat{R}, \check{R}$ in a decreasing order
```

```
Procedure 2 : Candidate Searching Mechanism()
 1: $\hat{D}^{REPL} \leftarrow 0$, $\widehat{sol} \leftarrow -1$, $vic \leftarrow -1$
 2: $\hat{D}^{TRAN} \leftarrow D^{pre}(i,k)$
 3: for $j \in V$ with breadth-first order from $i$ do
 4:     $\check{D}^{REPL} \leftarrow \min_{\forall u, I_{u,k}^{pre}=1}$ Network Simplex$(u, j, s^k)$
 5:     $\check{D}^{TRAN} \leftarrow$ Network Simplex$(i, j, r_{i,k})$
 6:     if node $j$ violates any capacity constraints then
 7:         $\check{D}^{TRAN} \leftarrow \check{D}^{TRAN} + \min_{\forall a, I_{j,a}^{pre}=1} D^{pre}(j,a)$
 8:     if $\check{D}^{REPL} + \check{D}^{TRAN} \leq \hat{D}^{REPL} + \hat{D}^{TRAN}$ then
 9:         $\hat{D}^{REPL} \leftarrow \check{D}^{REPL}$
10:         $\hat{D}^{TRAN} \leftarrow \check{D}^{TRAN}$
11:         $\widehat{sol} \leftarrow j$
12:         $vic \leftarrow a$
```

from requests in statistics; the other is the victim picking for application VMs when remaining resources of the candidate node is insufficient. For the first issue, the mechanism will calculate gains and losses so as to decide whether to replicate the VM to the candidate node (Line 4-5). Besides, we leverage the well-known method, network simplex algorithm [12], to do the calculation of minimum-cost flow and the parameters are represented as the source node, the destination node and the requirement of flow, respectively. On the other issue, when the candidate node is suffered from the resource shortage, the consideration should be more rigorous and comprehensive but not merely dealing with the tradeoff mentioned in the first issue. Therefore, the mechanism chooses the minimum demands of application from the candidate node (Line 6-7) and compare the traffic tradeoff caused between the victim application and the concerned application. If the replacement successfully reduces total network traffic, the VM re-allocation will be executed (Line 8-12). Otherwise, the mechanism keeps running breadth-first search to the next candidate node and calculates the traffic tradeoff again. Finally, our proposed scheme accumulates all network traffics and configure the placement of all replicas for required applications after all requests of different applications from different servers are processed.

**Theorem 2.** The time complexity of the proposed algorithm is $O((|V| + |E|)(|A||V|^2|E|^2\mathcal{C}\mathcal{U}))$.

*Proof:* In the beginning of the proposed scheme in Line 1-2, the priority setting mechanism will be triggered which needs to sort $\hat{R}$ and $\check{R}$ with total $|V||A|$ elements in a decreasing order and totally requires $O(|V||A| \log(|V||A|))$ time in the procedure 1. Next, the proposed scheme triggers the candidate searching mechanism for $r_{i,k} \in \hat{R}, r_{i,k} \in \check{R}$ in turn in Line 3-10. In the procedure 2, the candidate node will be traversed with the breadth-first order and the time complexity can be expressed as $O(|V| + |E|)$, since every node and every edge will be explored in the worst case. Moreover, the network simplex algorithm is called in Line 4-5, and the worst-case running time is $O(|V||E|^2\mathcal{C}\mathcal{U})$ [12], where $\mathcal{C}$ and $\mathcal{U}$ denote the largest arc cost and the largest arc capacity, respectively. Therefore, procedure 2 totally requires $O(|V||A|(|V| + |E|)(|V||E|^2\mathcal{C}\mathcal{U}))$ time. Consequently, the time complexity of Algorithm 1 is $O(|V||A|(\log(|V||A|) + (|V| + |E|)(|V||E|^2\mathcal{C}\mathcal{U}))) = O((|V| + |E|)(|A||V|^2|E|^2\mathcal{C}\mathcal{U}))$. ■

## IV. Performance Evaluation

### A. Simulation Setup

We develop a simulation model using C++ programs based on practical configurations to evaluate the performance of the proposed algorithm. Each fog node's capacity of computing, storage, and the application resource consumption for running a VM are based on the Amazon documents [13]. Besides, the network capacity of backhaul edges and the data rate requirement of applications are referred from the Cisco documents [14], [15]. The system initially generates the target users, each of who was randomly resided in the coverage area

TABLE II
Simulation Settings

| Symbol | Setting |
|---|---|
| Service time interval | 1000 units |
| Computing capacity of each fog node | 64 units |
| Storage capacity of each fog node | 64 GB |
| Edge capacity | 128 MB |
| Computing resource consumption of an application | [2, 8] units |
| Storage resource consumption of an application | [2, 8] GB |
| Data rate requirement of an application | [100, 800] KB |
| The maximum requests can be handled by a VM | 10 |
| User mobility pattern | Random-walk |
| Distribution of launching requests | Poisson-process |

TABLE III
Simulation Settings for Two Different Scales

| Symbol | Small-scale | Large-scale |
|---|---|---|
| Number of fog nodes | 10 | 50 |
| Number of edges | [20, 30] | [100, 150] |
| Number of users | 10 | 20 |
| Number of applications | 10 | 20 |

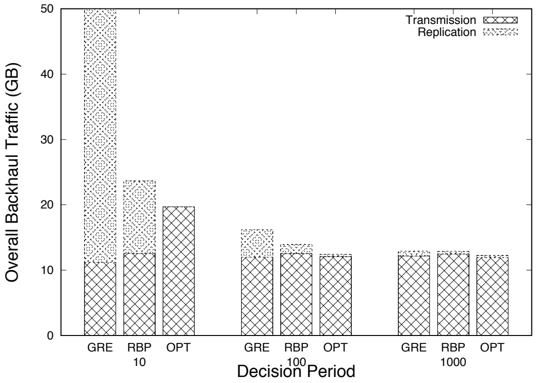

Fig. 2. The impacts of decision period on the overall backhaul traffic

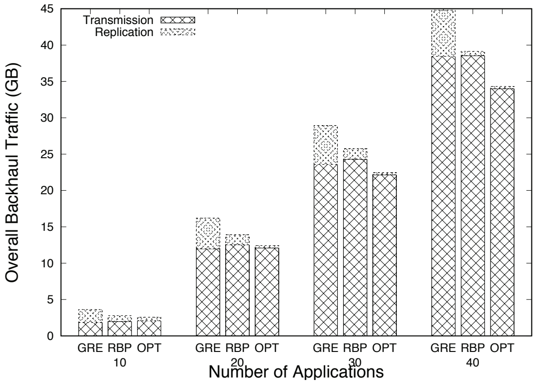

Fig. 3. The impacts of applications on the overall backhaul traffic

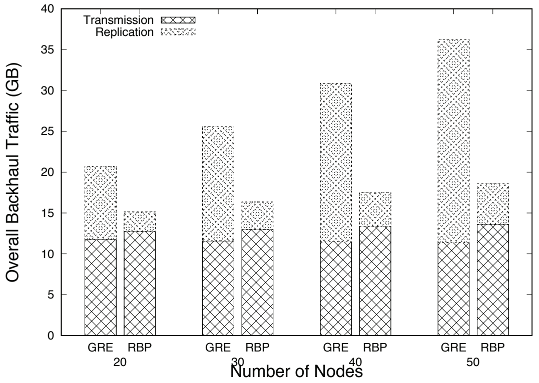

Fig. 4. The impacts of number of fog nodes on the overall backhaul traffic

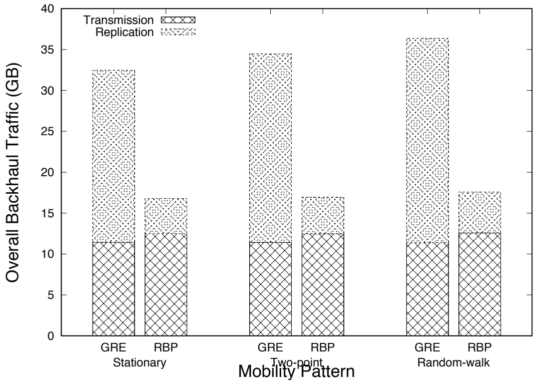

Fig. 5. The impacts of request distributions on the overall backhaul traffic

of different fog nodes while each fog node was also randomly deployed in the network.The simulation settings are listed in Table II.

We compare the proposed replication-based placement algorithm, denoted as *RBP*, with two algorithms. The first algorithm, denoted as *OPT*, uses the mixed integer linear programming tool provided by MATLAB to obtain an optimal solution [16]. The tool can find an optimal solution by using a branch and bound algorithm when the flow constraints of applications and users are released. The second VM placement algorithm, called greedy algorithm *GRE*, is designed for minimizing the data transmission traffic without consideration of VM traffic. Because *OPT* via the MATLAB tool can only be executed in a small scale network due to high computational complexity, we have two different scales of network settings as shown in Table III.

## *B. Simulation Results*

*1) In Small-scale:* Fig. 2 shows the impact of the decision period on the overall backhaul network traffic. As we can see, the overall traffic decreases as the decision period increases. This is because the algorithms placing each VM will consider minimizing the data transmission and the VM traffic during the decision period. When the decision period is longer, the VM traffic can be further reduced while data transmission traffic slightly increases under *GRE* and *RBP* because each VM will be moved at most once during the decision period. Moreover, under *OPT*, there is no VM traffic in decision period 10 because when the decision period is small, the VM traffic generated by the VM re-placement is more than the data transmission traffic. This result justifies our observation that the excessive and frequent VM re-allocation will bring serious overhead of VM replication. The simulation results show that our proposed algorithm can reduce at most more 223% overall traffic than *GRE*.

In Fig. 3, we demonstrate the impact of the number of applications on the overall backhaul network traffic. The overall traffic increases as the number of applications increases. The result is because more applications will have more application requests launched from users which will generate more data traffic. Moreover, the resource competition between applications is another main reason. When the number of applications increases, the number of VM replicas of each application may be cut down under the same environment. The fewer number of replicas, the farther routing distance between users and the corresponding VM. Our proposed algorithm considers the past statistic requests to minimize the overall traffic while *GRE* only attempts to minimize the data transmission traffic. Therefore, the simulation results show that our proposed algorithm can reduce at most more 30% overall traffic than *GRE*.

*2) In Large-scale:* Fig. 4 displays the impact of the number of fog nodes on the overall backhaul traffic. When the number of fog nodes increases, the total backhaul traffic increases, especially in terms of the VM traffic. This is because when there are more fog nodes, the VMs will be more scattered in order to reduce the data transmission traffic such that the VM traffic and the total traffic increases. We can see that compared with *GRE*, our proposed algorithm can reduce more VM traffic when the network scale is larger because our proposed algorithm will place each VM to serve as more user requests as possible. *RBP* can reduce at most more 94% overall traffic than *GRE*.

In Fig. 5, we show the impact of three different user mobility patterns, on the total backhaul traffic. *Stationary* mobility pattern represents all users stay under the coverage of the same fog nodes overtime, and *two-point* mobility pattern means that users will move back and forth over two different fog nodes. As shown in Fig. 5, when the user mobility is more irregular, the overall backhaul traffic is much higher. The reason is that for satisfying the application requests from irregularly moving users, *RBP* will consider the VM traffic to scatter replicas of applications to reduce the data transmission traffic while *GRE* will scatter replicas of applications without considering the VM traffic. Therefore, we can see that the overall backhaul traffic is mainly generated by the increase of the VM replication traffic when the user mobility is more irregular.

## V. Conclusion

In this paper, we have studied the VM placement problem in next-generation fog radio access networks. The objective is to minimize the overall backhaul traffic due to VM replication and data transmission. In this problem, we observe that the VM placement should not be frequently executed due to the huge amount of the VM migration traffic. We formulate the problem as an optimization problem and propose a replication-based VM placement algorithm which can be triggered in a long period. The simulations are conducted to justify our observation and show that compared with a greedy scheme, the proposed scheme can significantly reduce the overall backhaul traffic. The performance improvement is more evidence when the network scale is large and user mobility is irregular. Moreover, the simulation results show that the performance of the proposed algorithm is close to that of an optimal solution.

## Acknowledgment

This work was supported in part by Ministry of Science and Technology under Grant 105-2221-E-002-144-MY3 and Grant 105-2218-E-390-003-MY2, by Ministry of Education under Grant 105M7109, and by Information and Communications Research Laboratories of the Industrial Technology Research Institute (ICL/ITRI).

## References

[1] "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast, 2015-2020." [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.pdf

[2] "Akamai's [State of the Internet] Q1 2016 Report." [Online]. Available: https://www.akamai.com/us/en/multimedia/documents/state-of-the-internet/akamai-state-of-the-internet-report-q1-2016.pdf

[3] Y.-Y. Shih, W.-H. Chung, A.-C. Pang, T.-C. Chiu, and H.-Y. Wei, "Enabling Low-Latency Applications in Fog-Radio Access Network," *IEEE Network*, Jan. 2016.

[4] F. Bonomi, R. Milito, P. Natarajan, and J. Zhu, "Fog Computing: A Platform for Internet of Things and Analytics," *Big Data and Internet of Things: A Roadmap for Smart Environments*, vol. 546, pp. 169–186, 2014.

[5] G. Keller and H. Lutfiyya, "Replication and Migration as Resource Management Mechanisms for Virtualized Environments," in *Proc. of ICAS*, 2010, pp. 137–143.

[6] V. Medina and J. M. García, "A Survey of Migration Mechanisms of Virtual Machines," *ACM Comput. Surv.*, vol. 46, no. 3, Jan. 2014.

[7] X. Meng, V. Pappas, and L. Zhang, "Improving the Scalability of Data Center Networks with Traffic-aware Virtual Machine Placement," in *Proc. of IEEE INFOCOM*, 2010, pp. 1154–1162.

[8] T. Yapicioglu and S. Oktug, "A Traffic-Aware Virtual Machine Placement Method for Cloud Data Centers," in *Proc. of IEEE/ACM UCC*, 2013, pp. 299–301.

[9] S. Wang, R. Urgaonkar, M. Zafer, T. He, K. Chan, and K. K. Leung, "Dynamic Service Migration in Mobile Edge-Clouds," in *Proc. of IFIP Networking*, 2015, pp. 1–9.

[10] H. Yao, C. Bai, D. Zeng, Q. Liang, and Y. Fan, "Migrate or Not? Exploring Virtual Machine Migration in Roadside Cloudlet-based Vehicular Cloud," *Concurr. Comput. : Pract. Exper.*, vol. 27, pp. 5780–5792, Dec. 2015.

[11] A. Itai, "Two-Commodity Flow," *Journal of the ACM*, vol. 25, pp. 596–611, Oct. 1978.

[12] Z. Király and P. Kovács, "Efficient Implementations of Minimum-cost Flow Algorithms," *Informatica Acta Univ. Sapientiae*, vol. 4, pp. 67–118, 2012.

[13] "Amazon EC2 Instance Types." [Online]. Available: https://aws.amazon.com/ec2/instance-types/?nc1=f_ls

[14] "VNI Service Adoption Forecast - Services Gauge." [Online]. Available: http://www.cisco.com/c/en/us/solutions/service-provider/vni-service-adoption-forecast/vnisa_services_gauge.html

[15] M. Jaber, M. A. Imran, R. Tafazolli, and A. Tukmanov, "5G Backhaul Challenges and Emerging Research Directions: A Survey," *IEEE Access*, vol. 4, pp. 1743 – 1766, 2016.

[16] "Mixed-Integer Linear Programming Algorithms." [Online]. Available: http://www.mathworks.com/help/optim/ug/mixed-integer-linear-programming-algorithms.html