Mobile Networks and Applications (2019) 24:504–516
https://doi.org/10.1007/s11036-018-1153-5

State Management Function Placement for Service-Based 5G Mobile Core Architecture

Truong-Xuan Do[1] · Younghan Kim[1]

Published online: 24 October 2018

Abstract

Service-based 5G core architecture is designed to take advantages of network function virtualization and software-defined networking. In addition to the control and data plane separation, the service-based 5G core decouples the computing and storage resources which separates the mobile functions into two categories: "stateless" control functions and state management functions. Such new features improve 5G core network in terms of independent scalability and fast failure recovery. In geo-distributed cloud infrastructure-based 5G core networks, the "stateless" control functions can be deployed to all cloud centers close to access networks to reduce latency and traffic load burden. However, we can not deploy state management functions to all cloud centers which results in the high state transfer cost. In addition, we can not use only one state management function for entire network which results in high traffic load burden. Therefore, the placement of state management functions involves different conflicting design objectives which requires a new model to optimally place these functions. In this paper, we propose a multi-objective model which can achieve the balance between state transfer cost and traffic load burden on state management functions. We first solve our model using $\varepsilon - constraint$ approach which tries to optimize one objective while keeping another under threshold. Second, we propose an adaptive solution based on adaptive weighted sum approach to find a set of Pareto optimal solutions for our multi-objective model. Simulation results show that our proposed solution offers better balance between two design objectives compared to other solutions.

Keywords Service-based 5G core network · Unstructured data storage function · Stateless function · Network function virtualization

1 Introduction

Currently, three enabling technologies consisting of cloud computing, software-defined networking (SDN), and network function virtualization (NFV) have driven the evolution of mobile core network architecture. Looking back to history, when the cloud computing arrived, 4G evolved packet core (EPC) evolved from the dedicated hardware-based deployment to fully virtualized on cloud infrastructure(vEPC). When SDN [13, 24] came into play, the mobile core architecture is redesigned to take advantages of control plane (CP) and data plane (DP) separation concept. The control plane is implemented on the top of a centralized controller which can configure the data plane consisting of extended switches to process the encapsulation and decapsulation of mobile traffic. Such SDN-based mobile network enables efficient network control, operation, and programmable network. NFV [1] tries to deploy the mobile core functions as virtual network functions (VNF) on cloud infrastructure which makes the mobile core more scalable and reduces capital cost. NFV introduces a centralized management and orchestration framework [2] which highly automates the service deployment and management, as well as shortens the new service delivery time.

In order to take the advantages of SDN, 3GPP standardization development organization (SDO) redesigned the current 4G core architecture to the control user plane separation architecture (i.e. CUPS) [3]. However, in order to take fully advantages of cloud environment and NFV, 3GPP mobile network needs to be evolved one step more to service-based 5G architecture. The service-based 5G architecture [4, 5] redesigns the mobile network functions into more fine-grained network functions and microservices which are communicated with each other via application programmable interfaces (APIs). This new feature increases the reusability of mobile network functions which can be

✉ Truong-Xuan Do
xuan@dcn.ssu.ac.kr

[1] Soongsil University, Seoul, South Korea

orchestrated and shared among different network slices [18]. Beside control and data plane separation, the service-based 5G architecture introduces another layer of separation which decouples between computing and storage resources. The control plane functions are designed as “stateless” functions which can communicate with the separate state management functions (StateMF) for their internal processing state. These state management functions can store not just preconfigured data (e.g. subscriber profile and policy), but also processing state (e.g. UE contexts, forwarding context for ongoing sessions). These new design concepts will make the 5G mobile core functions more independently scalable and quickly recoverable. In other words, the service-based architecture makes 5G core more cloud-native and take fully advantages of NFV [25].

Two challenges should be addressed towards 5G core network deployment. The first challenge is to define service APIs among 5G core network functions, and the second challenge is to define new optimization models for planning the service-based core network architecture over geo-distributed cloud infrastructure. In service-based 5G core networks, “stateless” control functions and user plane functions can be deployed to all cloud centers close to access networks to reduce latency and traffic load burden. However, compare to “stateless” functions, the placement of state management functions is required to consider a new design goal specific to state management functions (i.e. minimize state transfer cost). This cost is caused when the UE handovers between two service areas which are managed by two different sets of state management functions. This design goal tries to deploy as least as possible the number of state management functions for whole network which can result in the high latency and heavy traffic load burden on each state management function. Therefore, the second design goal is to reduce traffic load burden on each state management function. This second goal tries to deploy as many as possible the state management functions on all cloud centers to fairly distribute traffic load. The problem is to find an appropriate model and optimal solution in terms of both design goals. In addition, the model should incorporate new feature on traffic model of 5G mobile users (i.e. 5G mobile users can request as many sessions as they want). In order to solve this problem, we propose a multi-objective optimization model that aims at finding optimal placement of state management functions over geo-distributed cloud infrastructure. We first solve our model by converting it into single objective models and use $\varepsilon-constraint$ method which tries to optimize one objective while keeping another under threshold. Second, we propose an adaptive Pareto optimal solution (i.e., APO), which is based on the adaptive weighted sum approach to find a set of Pareto optimal solutions. This adaptive approach is designed to perform more refinements which can help to obtain more Pareto optimal solutions. These Pareto optimal solutions can be selected to achieve the tradeoff between state transfer cost and traffic load burden. Therefore, this adaptive approach results in a better balance between two design goals compared to normal multi-objective model (i.e., with same weight factors) and single objective models. By varying handover frequency, session request rate, and user density, we evaluate and prove that our proposed adaptive approach can provide the most optimal solutions in terms of traffic load, state transfer cost, and required number of StateMF sets.

The remainder of our paper is structured as follows. Section 2 presents about the background and related works. Section 3, the placement problem is formulated. In Section 4, mathematical formulation and solutions are presented. Sections 5, 6 shows the numerical results and conclusion.

2 Background and related works

The 5G network structure consists of two main parts which are radio access networks (RAN) and core network (CN). First, we give the readers some background related to research topics in radio parts. Second, we present different approaches for core network architecture based on SDN and NFV. Third, modeling and optimization algorithms for these SDN and NFV-based mobile core networks are presented.

2.1 5G next generation radio networks

In term of resource management in 5G radio networks, the authors in [19] tackled the resource allocation problem while keeping the efficient energy consumption. The authors in [31], an optimization framework dealing with caching and resource sharing mechanisms to efficiently deliver contents to mobile users was proposed. In term of security, the authors in [20] proposed relay selection approaches to enhance security. In terms of specific use cases, the authors in [17] proposed a multi-radio 5G archiecture for connected and autonomus vehicles use case. For device-to-device (D2D) use case, the authors in [32] proposed a joint encoding rate allocation and description distribution optimization to enhance video streaming performance over D2D communication in 5G networks.

2.2 SDN and NFV-based mobile core network and evolution to service-based architecture

In academia, with regard to SDN, the authors in Softcell [34], MobileFlow [26] proposed to deploy mobile network functions on the top of SDN controllers and use Openflow switches for the data traffic forwarding. In [11], authors discussed security issues for software-defined mobile networks. In [27], authors presented a testbed implementation of software-defined architecture for managing and monitoring wireless networks. In our previous work, we proposed a SDN-based architecture to support multicast and broadcast services in mobile core network [12]. Considering NFV, the authors in [6] presented a fully virtualized mobile core architecture over cloud infrastructure. In [15], software-as-a-service approach is presented for virtual mobile core. For more background about SDN and NFV-based mobile core network, interested readers can refer two comprehensive surveys [21, 22]. For standardization activities, the 3GPP SDO also takes advantages of SDN concept by introducing the CUPS architecture [3] which focuses on the separation of control and data plane of mobile gateways (i.e. S-GW and P-GW).

However, the current CUPS architecture is not flexible enough to leverage the strengths of NFV. As a result, the 3GPP 5G mobile core network evolves to service-based architecture [4, 5]. The architecture of service-based 5G mobile core is shown in Fig. 1. Three are three layers in this architecture: state management layer, control layer, and data forwarding layer. State management layer includes state management functions, such as network repository function (NRF), unstructured data storage function (UDSF), user data repository (UDR). These state management functions store both preconfigured data (e.g. UDR stores subscriber and policy data) to ongoing session-specific state information (e.g. UDSF stores UE context, forwarding state, and session context). The control layer includes “stateless” control plane functions, such as authentication server function (AUSF), access mobility function (AMF), session management function (SMF), policy charging function (PCF). The data forwarding layer includes user plane functions (UPF). In our model, we focus on the placement problem of the state management functions.

2.3 Placement problems of mobile core network functions

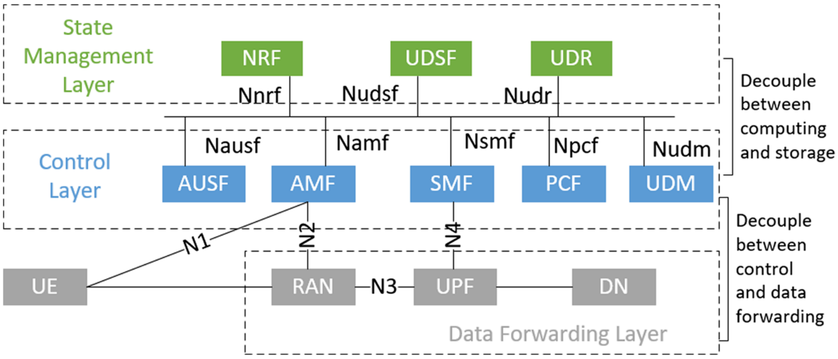
Along with the evolution of mobile core network, different optimization models have been developed for optimally placing mobile network functions. These models could be classified into two areas: (a) placement of control and data plane (i.e. controllers and switches) in the SDN-based architecture and (b) placement of virtual mobile functions in the NFV-based architecture.

The placement problems of SDN controllers and switches in SDN-based architecture have been introduced in [16] which proposed a heuristic algorithm considering control plane latency and resilience aspects. In [23], the authors proposed a controller placement algorithm considering both control latency and controller load. In [8], four deployment options for S-GW and P-GW which are decomposed into control and user plane functions are considered. The authors tried to optimize transport network load against data plane delay and number of potential data centers.

For the placement of virtual mobile functions in the NFV-based architecture, in [28], authors proposed an optimal placement solution for virtual core gateways to cope with the growth of traffic in case of large crow events. In [29], the authors proposed a VNF placement solution for creating a Serving-Gateway (S-GW) over federated clouds so that the S-GW relocation frequency is minimized. In [7], the authors considered application type and service requirements as metrics for creating VNF instances of Packet-Gateway (P-GW) and introduced three heuristic solutions to deal with the problem. In [30], the authors formulated and solved a multi-objective optimization problem which optimizes packet delivery path and S-GW relocation for optimally placing VNF instances of S-GW and P-GW. Three solutions based on game theory were proposed and evaluated with mobility feature and traffic pattern. In [10], an optimization model for link and node capacity has been proposed for placing the virtual mobile core functions. In [9], authors considered jointly both SDN and NFV by taking into account two objective functions which are network load cost and data center resources cost. Their optimization models included the requirements for both control and data plane latency as well as the number of data centers.

However, none of existing related works on modeling and optimizing the placement of mobile network functions covered the multi-objective model between state transfer

Fig. 1 Service-based 5G mobile network architecture



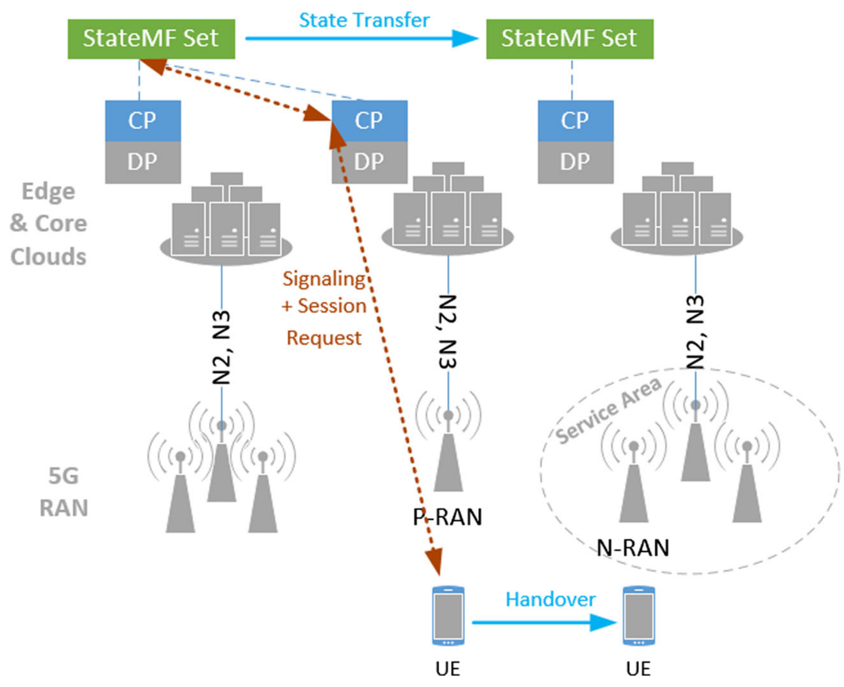


Fig. 2 Deployment architecture for service-based 5G mobile network

cost and traffic load burden, as well as new feature on traffic model of 5G mobile users to find the optimal locations for state management functions over geo-distributed cloud infrastructure.

3 Problem formulation

In this section, we present the deployment architecture and placement problem of state management functions over a federated cloud infrastructure. As shown in Fig. 2, we assume that the mobile network operator owns a number of cloud centers (namely a federated cloud infrastructure) which could be edge clouds or core clouds distributed over different geographical locations. Each cloud location can host one or several stateless CP and DP functions which are used to serve a predefined region of 5G RAN nodes via N2 and N3 interfaces, namely service area. These CP and DP functions can be scaled in or out depending on the UE density and traffic amount to be carried out over the corresponding service area. Each service area is designated to connect to the set of CP and DP functions on the corresponding cloud center. Each set of CP functions is designated to use one set of state management functions(StateMF set). Our problem is to determine the number of StateMF sets needed to deploy and their optimal locations over cloud infrastructure. The simplified procedures for initial registration, handover, and new PDU session request is depicted in Fig. 3. In every operation of mobile users, the mobile users (i.e. UE) always have to access to a StateMF set. Based on our observation, when UE performs handover from previous radio access node (P-RAN) in one service area to next radio access node (N-RAN) in another service area, if two service areas are managed by two different StateMF sets, the state transfer occurs. The state information (e.g. UE context, forwarding context) required to maintain the connection session between UE and network has to be transferred to current set of serving StateMFs as depicted in Fig. 3. This state transfer results in high handover latency and signaling overhead and should be avoided. Therefore, the first goal in placing these StateMFs is to minimize as much as possible the state transfer among different StateMF sets. In order to achieve this objective, we need to reduce the number of StateMF sets and try to push traffic load to one specific StateMF set. However, this approach will create traffic load burden on StateMFs on one cloud center. Therefore, the second goal in placing these StateMFs is to minimize the traffic load going to each StateMF sets on each cloud center. However, in order to minimize the traffic load, we need to deploy more StateMF sets and balance the traffic load to the StateMF sets on different cloud centers. This leads to the increment of state transfer when the UE moves among service areas. Therefore, we need to solve multi-objective optimization problem to find the trade-off solution between traffic load and state transfer.

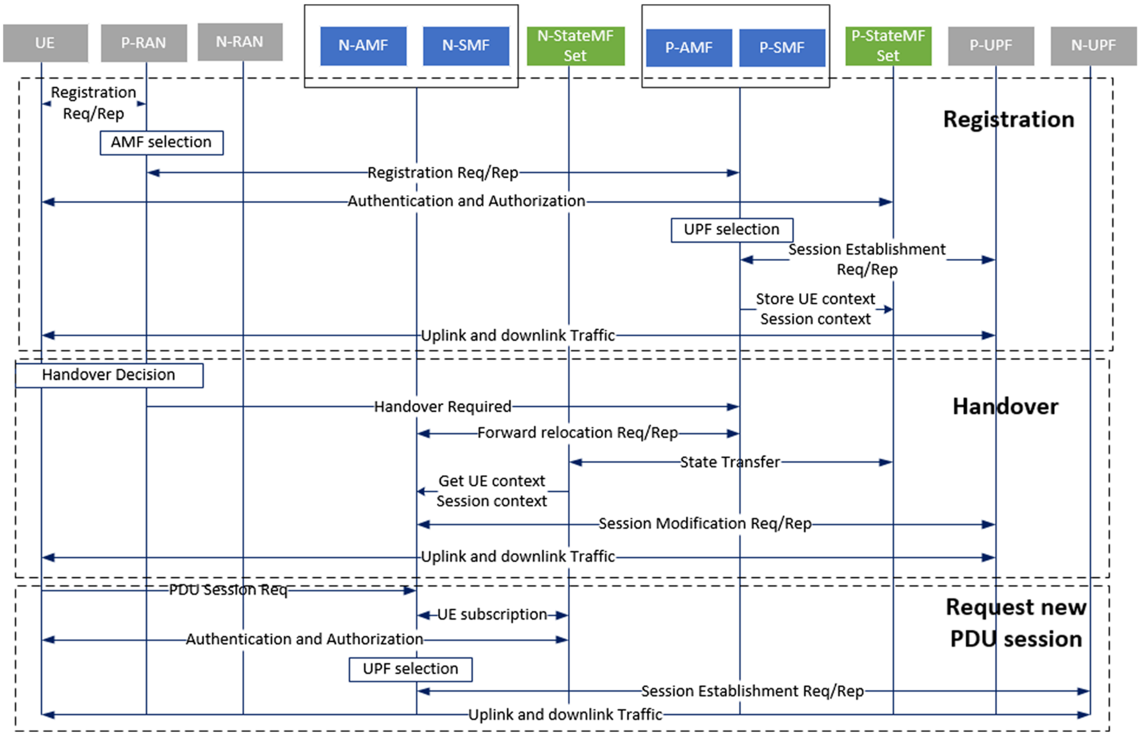

UE
P-RAN
N-RAN
N-AMF
N-SMF
N-StateMF Set
P-AMF
P-SMF
P-StateMF Set
P-UPF
N-UPF
Registration Req/Rep
AMF selection
Registration
Registration Req/Rep
Authentication and Authorization
UPF selection
Session Establishment Req/Rep
Store UE context Session context
Uplink and downlink Traffic
Handover Decision
Handover
Handover Required
Forward relocation Req/Rep
State Transfer
Get UE context Session context
Session Modification Req/Rep
Uplink and downlink Traffic
PDU Session Req
Request new PDU session
UE subscription
Authentication and Authorization
UPF selection
Session Establishment Req/Rep
Uplink and downlink Traffic

Fig. 3 Simplified procedures for service-based 5G network

4 Solution description

We make assumptions that the network operator possesses M cloud centers and M corresponding service areas (denoted as S_i). Let $h(i, j)$ denote the handover frequency between areas i and j. Let n_i denote the average number of UEs and u_i denote the average session request rate generated by each UE in the service area S_i. For simplicity, we assume that L_{sig} denotes the traffic load going to the StateMF set on one cloud center due to procedures such as registration and handover. We denote $L_{session}$ as the traffic load going to one StateMF set when the UE requests a Protocol Data Unit (PDU) session. We denote β as the cost for one state transfer. We define our decision variables as two matrices: X and Y. If two service areas S_i and S_j use the same StateMF set, $X(i, j) = 1$, otherwise $X(i, j) = 0$. If S_i is controlled by StateMFs on the cloud center t, $Y(i, t) = 1$, otherwise $Y(i, t) = 0$. Our placement problem is formulated as the following integer linear program (Table 1):

Our two main goals are: (i) optimize the state transfer cost among sets of StateMFs which are shared among different service areas (ii) optimize the traffic load going to each StateMF set deployed on each cloud center. The traffic usage model of mobile users in service-based 5G network is different from that in 4G network. In service-based 5G network, the UE can request multiple PDU sessions to

Table 1 State management function cluster placement

Notation	Description
$X(i, j)$	Equal 1 if two service areas use the same StateMF set
$Y(i, t)$	Equal 1 if service are i controlled by StateMF on cloud center t
M	Number of cloud centers and service areas
$h(i, j)$	Handover frequency between areas i and j
n_i	Average number of UEs in service area i
u_i	Average number of session requests by each UE in service area i
L_{sig}	Traffic load going to the StateMF set due to registration procedure
$L_{session}$	Traffic load going to the StateMF when UE requests a PDU session
β	Unit cost for state transfer

multiple data networks at the same time. Therefore, the session request rate of UEs also affects the traffic load on the StateMFs.

$$\textbf{Minimize} \quad \sum_{i\in M}\sum_{j\in M}\beta h(i,j)(1-X(i,j)) \tag{1}$$

$$\textbf{Minimize} \quad \forall t \in M : \sum_{i\in M} Y(i,t)n_i(L_{sig}+u_i L_{session}) \tag{2}$$

Meanwhile, the constraints for linear programming are defined as followings:

1) Constraint guarantees that the matrix $X(i,j)$ is symmetric

$$\forall i \in M, \forall j \in M : X(i,j) = X(j,i) \tag{3}$$

2) Constraint guarantees that the matries X and Y are binary

$$\forall i \in M, \forall j \in M : X(i,j) \in [0,1] \tag{4}$$

$$\forall i \in M, \forall t \in M : Y(i,t) \in [0,1] \tag{5}$$

3) Constraint guarantees that if $X(i,j) = 0$, two service areas shouldn't use the same StateMF set on one cloud center

$$\forall i \in M, \forall j \in M, \forall t \in M : Y(i,t)+Y(j,t) \leq 1+X(i,j) \tag{6}$$

4) Constraint guarantees that one service area is at least connected to one cloud center

$$\forall i \in M : \sum_{t\in M} Y(i,t) = 1 \tag{7}$$

5) Constraint guarantees that if $X(i,j) = 1$, two service areas should use the same StateMF set on one cloud center

$$\forall i \in M, \forall j \in M, \forall t \in M : |Y(i,t)-Y(j,t)| \leq 1-X(i,j) \tag{8}$$

6) Constraint guarantees that the StateMFs should be deployed in redundancy to ensure the availability

$$\sum_{t\in M}\sum_{i\in M}(1-X(i,j)) \geq 1 \tag{9}$$

Here, to solve the multi-objective optimization problem, we first present two simple solutions which try to convert the multi-objective problem into single objective problem and solve them using linear programing solvers. Next, we present our adaptive multi-objective approach to find Pareto optimal solutions which can achieve the balance between two design objectives.

4.1 Optimize state transfer cost (OST)

In this solution, the $\varepsilon - constraint$ approach is used to minimize the state transfer cost among StateMF sets. This approach will fix the upper boundary for one objective function and try to optimize the another. Here, we denote $TrafficLoad_{max}$ as the maximum traffic load on the StateMFs on one cloud center. The optimization model which targets at optimizing the state transfer cost among StateMF sets can be formulated as the following integer linear program.

$$\textbf{Minimize} \quad F(X,Y) = \sum_{i\in M}\sum_{j\in M}\beta h(i,j)(1-X(i,j)) \tag{10}$$

subjectto

$$\forall i \in M, \forall j \in M : X(i,j) = X(j,i) \tag{11}$$

$$\forall i \in M, \forall j \in M : X(i,j) \in [0,1] \tag{12}$$

$$\forall i \in M, \forall t \in M : Y(i,t) \in [0,1] \tag{13}$$

$$\forall i \in M, \forall j \in M, \forall t \in M : Y(i,t)+Y(j,t) \leq 1+X(i,j) \tag{14}$$

$$\forall i \in M : \sum_{t\in M} Y(i,t) = 1 \tag{15}$$

$$\forall i \in M, \forall j \in M, \forall t \in M : |Y(i,t)-Y(j,t)| \leq 1-X(i,j) \tag{16}$$

$$\sum_{i\in M}\sum_{j\in M}(1-X(i,j)) \geq 1 \tag{17}$$

$$\forall t \in M : \sum_{i\in M} Y(i,t)n_i(L_{sig}+u_i L_{session}) < TrafficLoad_{max} \tag{18}$$

where $TrafficLoad_{max}$ represents the maximum load supported by one StateMF set deployed on one cloud center. Constraint (18) allows to maintain the acceptable load on one StateMF set which could be fixed by network operator.

4.2 Optimize traffic load (OTL)

In the second solution, we also use $\varepsilon - constraint$ approach. We target at minimizing the traffic load going to one StateMF set on one cloud center while keeping the state transfer cost under an acceptable threshold. We denote that $StateTransferCost_{max}$ as the maximum state transfer cost in whole mobile network. We formulate the same integer linear program as previous model.

Minimize

$$\forall t \in M : G(X,Y) = \sum_{i \in M} Y(i,t) n_i (L_{sig} + u_i L_{session}) \tag{19}$$

subjectto

$$\forall i \in M, \forall j \in M : X(i,j) = X(j,i) \tag{20}$$

$$\forall i \in M, \forall j \in M : X(i,j) \in [0,1] \tag{21}$$

$$\forall i \in M, \forall t \in M : Y(i,t) \in [0,1] \tag{22}$$

$$\forall i \in M, \forall j \in M, \forall t \in M : Y(i,t) + Y(j,t) \leq 1 + X(i,j) \tag{23}$$

$$\forall i \in M : \sum_{t \in M} Y(i,t) = 1 \tag{24}$$

$$\forall i \in M, \forall j \in M, \forall t \in M : |Y(i,t) - Y(j,t)| \leq 1 - X(i,j) \tag{25}$$

$$\sum_{i \in M} \sum_{j \in M} (1 - X(i,j)) \geq 1 \tag{26}$$

$$\sum_{i \in M} \sum_{j \in M} \beta h(i,j)(1 - X(i,j)) < StateTransferCost_{max} \tag{27}$$

First six constraints are the same as previous model, a new constraint (27) is defined to keep maintaining the state transfer cost at the acceptable level.

4.3 Adaptive trade-off solution between traffic load and state transfer cost (APO)

In this solution, we try to figure out a trade-off solution between two objectives: state transfer cost (i.e., $F(X,Y)$) and traffic load (i.e., $G(X,Y)$). These two objectives conflict to each other, so it's difficult to find an optimal solution for both objectives at the same time. Therefore, Pareto optimal set, including Pareto optimal solutions will be solution to this kind of multi-objective obtimization problem. A solution is called Pareto optimal if none of the objective functions can be improved in value without degrading other objective values. As depicted in Fig. 4, the Pareto optimal solutions are the concepts in decision space and become Pareto optimal front in objective space.

In order to derive the Pareto optimal solutions, we propose an adaptive Pareto optimal approach (APO) based on the adaptive weighted sum approach [33]. This approach can help find more Pareto optimal points on the Pareto optimal front than normal weighted sum approach. The APO approach is depicted in Algorithm 1. We define a function $optimizePareto(F_{lb}, G_{lb}, F_{ub}, G_{ub}, w)$. This function takes input parameters as upper bounds (i.e., F_{ub}, G_{ub}) and lower bounds (i.e., F_{lb}, G_{lb}) of both objective functions, as well as weight factor w. This function will solve a multi-objective model using normal weighted sum method to produce one optimal solution between upper bounds and lower bounds of two objective functions. Here, a solution mean a point on Pareto optimal front in objective space which can be easily converted back to a Pareto optimal solution in decision space. Because these two objective functions do not have the same units, so the normalization is required before applying weight factor. The multi-objective model needed to solve after normalization and applying weight factor is presented as follows.

Minimize

$$w * \frac{F(X,Y) - F_{lb}}{F_{ub} - F_{lb}} + (1-w)\frac{G(X,Y) - G_{lb}}{G_{ub} - G_{lb}} \tag{28}$$

subjectto

$$\forall i \in M, \forall j \in M : X(i,j) = X(j,i) \tag{29}$$

$$\forall i \in M, \forall j \in M : X(i,j) \in [0,1] \tag{30}$$

$$\forall i \in M, \forall t \in M : Y(i,t) \in [0,1] \tag{31}$$

$$\forall i \in M, \forall j \in M, \forall t \in M : Y(i,t) + Y(j,t) \leq 1 + X(i,j) \tag{32}$$

$$\forall i \in M : \sum_{t \in M} Y(i,t) = 1 \tag{33}$$

$$\forall i \in M, \forall j \in M, \forall t \in M : |Y(i,t) - Y(j,t)| \leq 1 - X(i,j) \tag{34}$$

$$\sum_{i \in M} \sum_{j \in M} (1 - X(i,j)) \geq 1 \tag{35}$$

$$\sum_{i \in M} \sum_{j \in M} \beta h(i,j)(1 - X(i,j)) < F_{ub} \tag{36}$$

$$\forall t \in M : \sum_{i \in M} Y(i,t) n_i (L_{sig} + u_i L_{session}) < G_{ub} \tag{37}$$

In Algorithm 1, we first solve two previous single objective models in Sections 4.1 and 4.2. The output of these two single objective models are the best and worst values of both objective functions, denoted as F_{best}, G_{best}, F_{worst}, G_{worst}. We obtain initial solutions by calling the function $optimizePareto()$ with upper and lower bounds set to $F_{best}, G_{best}, F_{worst}, G_{worst}$. The weight factor w runs from 0 to 1 with the uniform step size $\frac{1}{n_{init}}$. n_{init} is a number of divisions, initially set to find a number of initial solutions. The initial solutions are stored in $paretoSetInit$. A lot of solutions are overlapped when the normal weighted sum is used. Therefore, we try to find more solutions between two initial consecutive solutions in $paretoSetInit$. We calculate the number of further refinements ref_i for each segment between two consecutive solutions. The longer segment, the more refinement steps we should run. The number of refinement steps are calculated as $ref_i = round(\frac{length(segment)}{length(shortestsegment)}) * C$ where C is a constant of algorithm, chosen by experiment

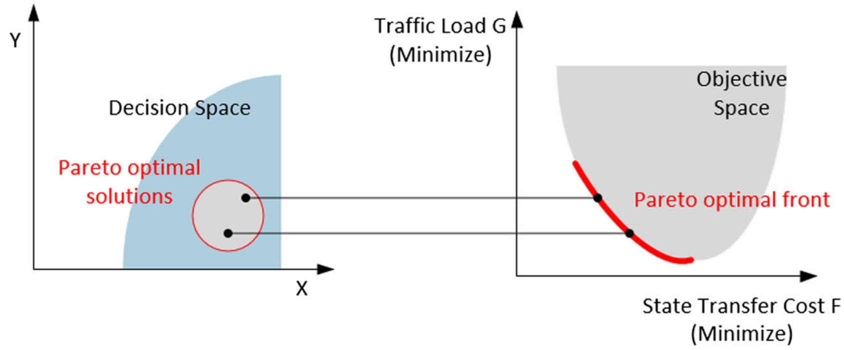


Fig. 4 Pareto optimal solutions

to produce as many as possible the Pareto optimal solutions. We also limit ref_i less than 20 to avoid algorithm to run too long in case of $shortestsegment$ is much shorter than $segment$. For each segment, we run again the function $optimizePareto()$ to find more solutions. The upper and lower bounds are set to the two endpoints of each segment. The weight factor w runs from 0 to 1 with the step size $\frac{1}{ref_i}$. For each feasible model, one Pareto solution will be obtained.

Algorithm 1 Adaptive Pareto optimal approach (APO)

```
Input: $M, h, u, n, L_{sig}, L_{session}, n_{init}$
Begin:
$paretoSet \leftarrow \emptyset$
$F_{best}, G_{worst} \leftarrow$ minimize $F(X, Y)$
$F_{worst}, G_{best} \leftarrow$ minimize $G(X, Y)$
for $w = 0 : \frac{1}{n_{init}} : 1$ do
    solve multi-objective model $optimizePareto()$
    with input $F_{best}, G_{best}, F_{worst}, G_{worst}, w$
    if model is feasible then
        add solution $(F, G)$ into $paretoSetInit$
        add solution $(F, G)$ into $paretoSet$
    end if
end for
calculate segment length between two consecutive $(F, G)$
calculate further refinement for each segment $ref_i$
for $i = 0 : 1 : (length(paretoSetInit) - 1)$ do
    for $w = 0 : \frac{1}{ref_i} : 1$ do
        solve multi-objective model $optimizePareto()$
        with input $F_i, G_{i+1}, F_{i+1}, G_i, w$
        if model is feasible then
            add point $(F, G)$ values into $paretoSet$
        end if
    end for
end for
Finish
Output: $paretoSet$
```

5 Evaluation

5.1 Simulation setup and parameters

In order to evaluate solutions obtained from our proposed multi-objective model and single objective models, we developed a simulator program using Python and Gurobi optimization library [14] for integer linear programming. The simulations are run on a server which uses Intel(R) Xeon(R) CPU D-1540 @ 2.00GHz and 64 GB memory. First, we run the Algorithm 1 to find Pareto optimal front in the objective space with different scenarios. Second, we compare solutions obtained from single objective models (i.e., OST and OTL), normal multi-objective model (i.e., PO with same weight factors for both objectives), and proposed adaptive multi-objective model (i.e., APO) in terms of the following metrics:

- **State transfer cost**: the cost for state transfer generated when the UEs handover.
- **Maximum traffic load**: the maximum traffic load going to sets of StateMFs among cloud centers.
- **The number of StateMF sets**: is equivalent to the number of StateMF sets required to deploy over cloud centers.

Simulation parameters are assumed like this: the number of cloud centers and service areas M are set to 10. The handover frequency $h(i, j)$ between areas i and j are uniformly distributed between 10 and 100. The average number of UEs n_i in a service area are uniformly distributed between 100 and 1000. The average number of session requests u_i are uniformly distributed between 1 and 10. The traffic load on the StateMF due to registration procedure L_{sig} and session request procedure $L_{session}$ are set to 10. The initial number of divisions n_{init} is 10 and the constant C is set to 10. The unit cost for state transfer β is set to 1 unit. The maximum traffic load $TrafficLoad_{max}$ and state transfer cost $StateTransferCost_{max}$ are set to 250000 and 7000, respectively.

5.2 Pareto optimal selection for multi-objective model

We run Algorithm 1 to find possible Pareto optimal front each scenario. For each scenario, the multi-objective model is solved with the weight factor runnning between [0, 1] to obtain initial solutions. Then, the refinement process is executed for each segment between two consecutive solutions. The multi-objective model continues to be solved for each segment to find more Pareto optimal solutions. In the first scenario, we fix the average number of UEs (i.e., 550) and handover frequency (i.e., 50). We evaluate the algorithm for the number of session requests among (2, 4, 6). From Fig. 5a, with the number of requests equal 2 or 4, the most optimal solution can be obtained at $w = 0.5$ which balances between two design objectives. However, with the number of requests equal 6, the weight factor $w = 0.5$ does not provide the balance between the state transfer cost and traffic load. The traffic load at $w = 0.5$ is too high and close to the worst traffic load at $w = 1$. The more balance solution is shown in Fig. 5a, which is obtained by doing more refinements between two initial solutions at $w = 0.4$ and $w = 0.5$. In the second scenario, we fix the average number of UEs (i.e., 550) and number of session requests (i.e., 5). We evaluate the algorithm for the handover frequency among 20, 40, 80. From Pareto optimal fronts shown in Fig. 5b, we can observe that the best balance solutions for different handover frequencies are below the solution at $w = 0.5$. These solutions can be obtained by further refinements between two solutions at $w = 0.5$ and $w = 0.4$. In the third scenario, we fix the handover frequency (i.e., 50) and number of session requests (i.e., 5). We evaluate the algorithm for the number of UEs among 150, 550, 950. From Fig. 5c, we also see that for number of UEs equal 150, the solution at $w = 0.5$ can be the most balanced. For the number fo UEs equal 550, the best balance

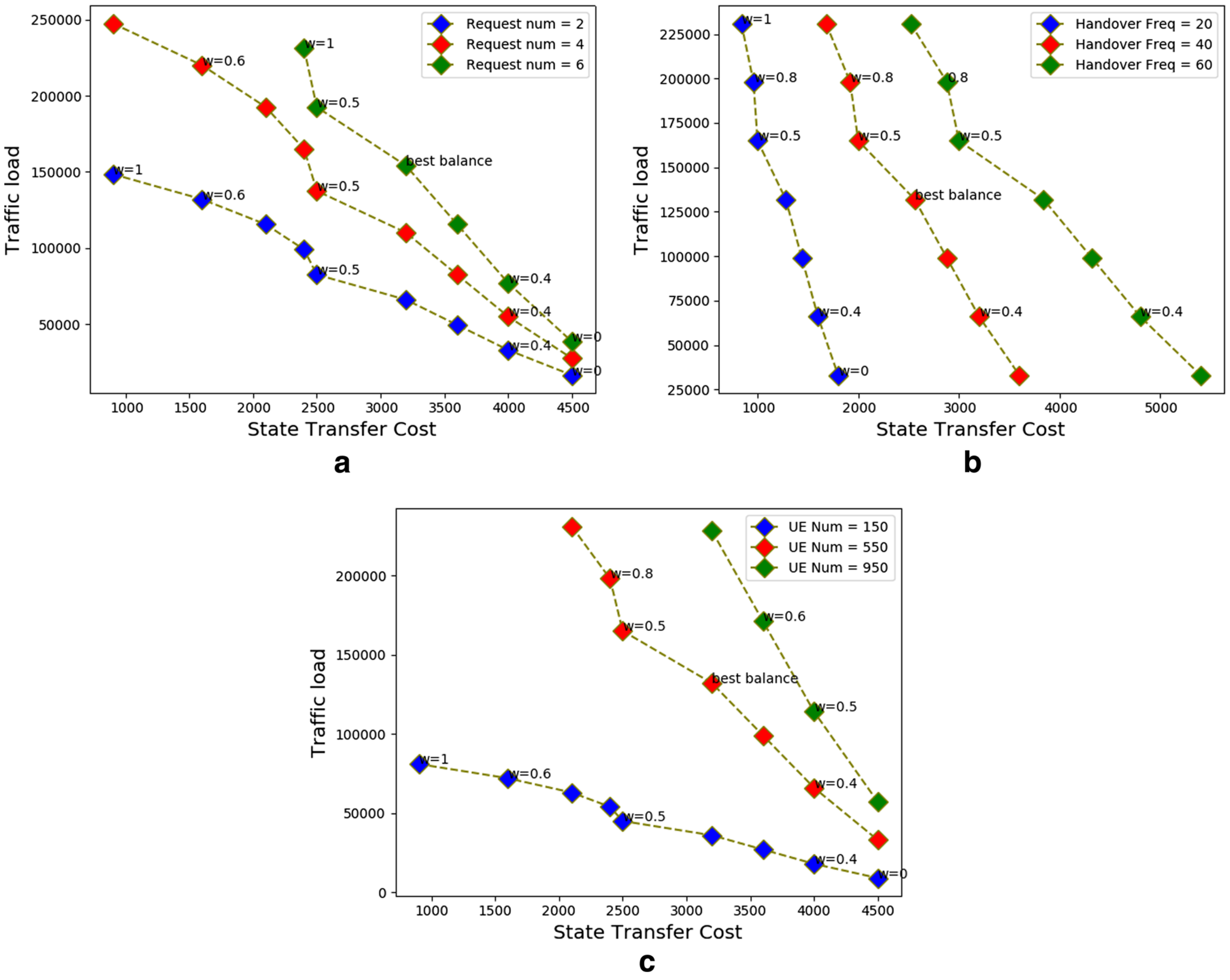


Fig. 5 Pareto optimal front for multi-objective model

solution is below the solution at $w = 0.5$. The best balance solution in case of 950 can be at $w = 0.6$ or $w = 0.5$. It is worth mentioning that here the mobile operator can select a different Pareto solution depending on real values of state transfer cost and traffic load. Our proposed multi-objective model can provide the network operators with the possible solutions to design an optimal network which balances between two design objectives.

5.3 Performance results

Figures 6, 7, and 8 shows the performance of our proposed adaptive model (i.e., APO), normal PO (i.e., PO with same weight factors $w = 0.5$), and two single objective models (i.e., OST and OTL) when we vary handover frequency, number of session requests, and number of UEs. Generally, in all cases, we can see that the OST outperforms two others in terms of state transfer cost and required number of StateMF sets due to the OST tries to optimize state transfer and use as small number of StateMF sets as possible to reduce state transfer when UEs handover. The OTL is better than two others in term of traffic load due to the approach tries to optimize the traffic load on StateMF sets. The normal PO can achieve the balance between the state transfer and traffic load in some scenarios. However, in some other scenarios, it is biased to one side. In most scenarios, the APO can achieve the balance between the two design objectives.

From Fig. 6a and b, we can observe that when the handover frequency increases, the state transfer cost of all solutions also increase accordingly. The state transfer cost of OST is still lower than two others due to the objective of OST is to minimize the state transfer cost. The traffic load and number of StateMF sets of OST are constant which is due to the fact that this approach tries to use as small amount of StateMF sets as possible but still satisfies the maximum

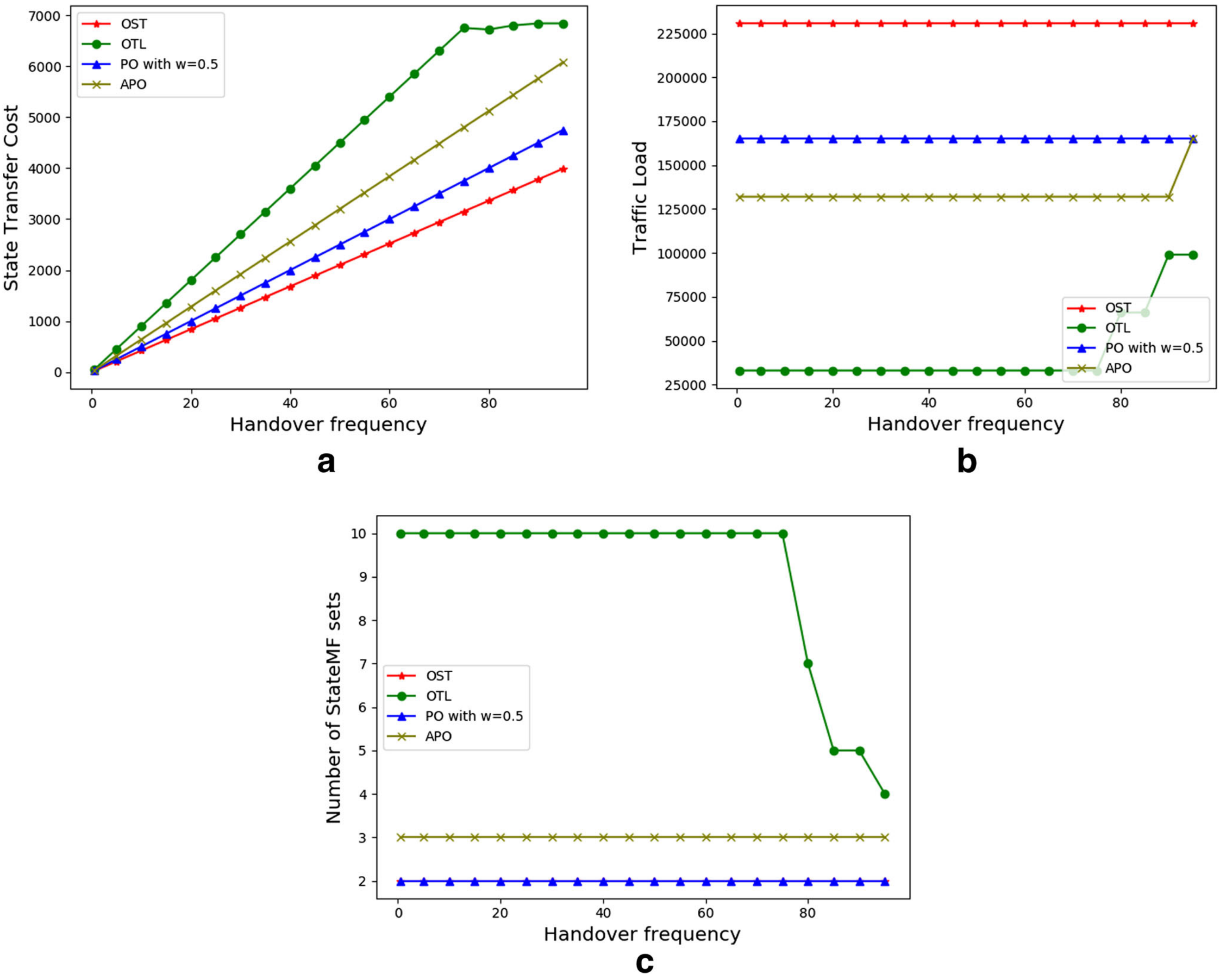


Fig. 6 Performance comparison of proposed solutions as the variation of handover frequency

traffic load requirement. This traffic load requirement is not affected by the handover frequency variation. In contrast, when the handover frequency reaches to a certain point (i.e 75), the traffic load of OTL start increasing which is due to the fact that the maximum state transfer cost requirement is affected. It makes the number of StateMF sets decrease to satisfy the maximum state transfer constraint. We can see that the normal PO is biased to state transfer cost more than traffic load in most values of handover frequency. Our proposed adaptive approach APO offers better balance but it comes at the expense of higher number of StateMF sets as shown in Fig. 6c.

From Fig. 7a and b, we observe that when the number of session requests is less than 4, the normal PO can achieve the good balance between state transfer cost and traffic load. However, the number of session requests is greater than 4, the maximum traffic load constraint is affected, which results in the increasement of state transfer cost of the OST. In this scenario, the normal PO is biased to the OST approach when the number of session requests is less than 6 and biased to the OTL when the number of session requests is greater than 6. The proposed APO provides better Pareto optimal solutions in the most of values of number of session requests compared to the normal PO.

Similarly, Fig. 8 shows that the APO always offers most balanced solution compared to other solutions in terms of both state transfer cost and traffic load with the adequate number of StateMF sets.

5.4 Discussion on real-time deployment

Figure 9 presents the running time of different models (i.e., OST, OTL, PO, and APO) in log scale. The APO is more time-comsuming than three other approaches. This is because of the APO needs to solve the multi-objective model many times with different weight factors and several

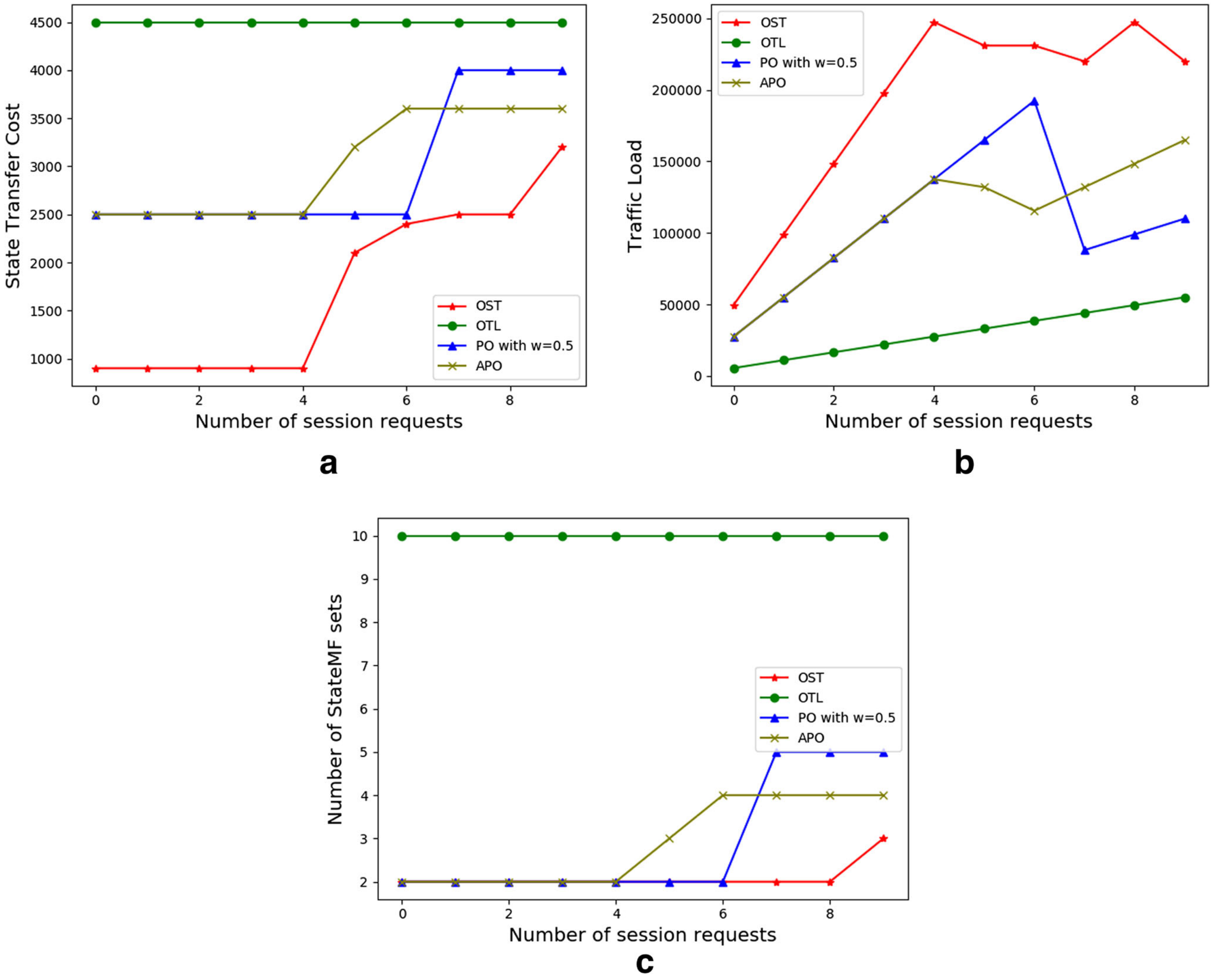


Fig. 7 Performance comparison of proposed solutions as the variation of number of session requests

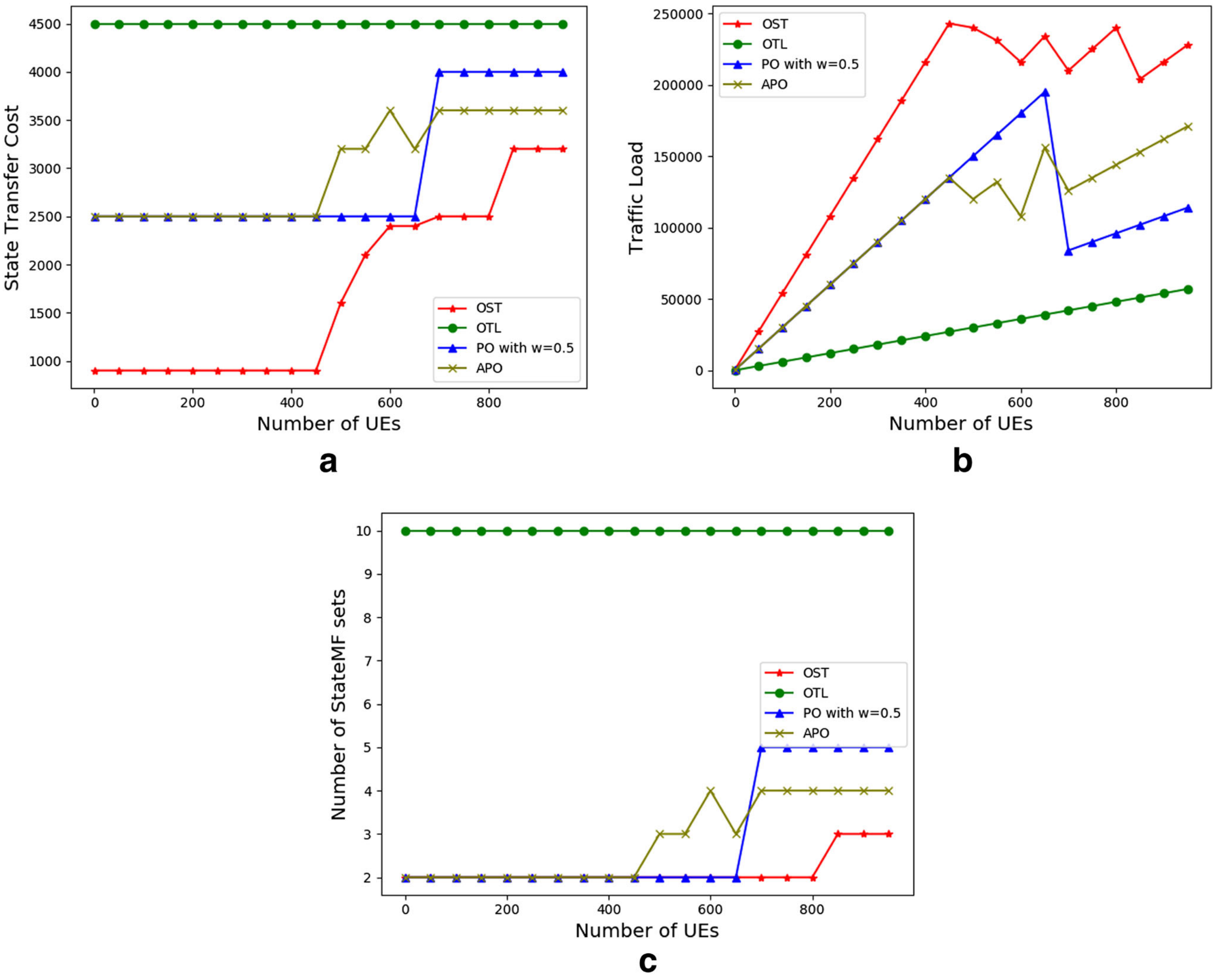


Fig. 8 Performance comparison of proposed solutions as the variation of user density

refinements to find all possible Pareto optimal solutions. As the problem size increases, the running time for the APO also increases quite a lot. However, in the real life scenario, the state management functions do not need to be deployed in a real-time manner. These functions are normally deployed at the initial stage of network planning. In addition, we can tune the number of refinements (i.e., C and n_{init}) which can result in less optimal solutions but reduce the running time when the real-time deployment is required.

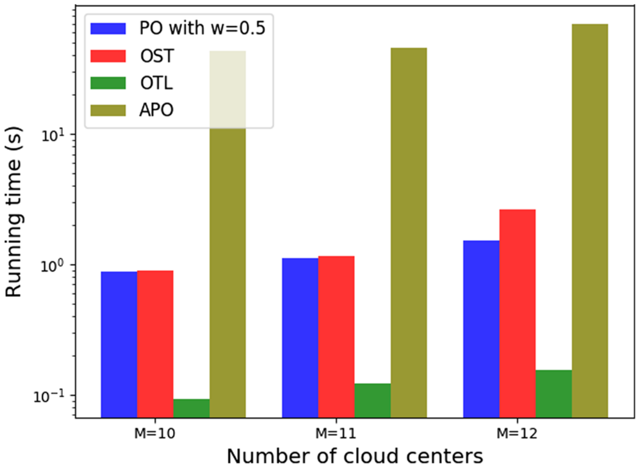


Fig. 9 Running Time

6 Conclusion

In this paper, we formulate a multi-objective optimization problem for placing state management functions in service-based 5G mobile core networks over geo-distributed cloud infrastructure. Our problem takes into account two objective functions, including traffic load processed by each set of state management functions and state transfer cost. We proposed an adaptive multi-objective approach to this problem which was proved by simulation that it can provide optimal solutions for our problem under various network conditions. In our future work, we will consider more constraints into our problem (e.g. latency) to give the best placement solutions in another use case (e.g., industrial control) which has strict requirement on network latency.

Acknowledgment This work was supported by Institute for Information and communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (No. IITP-2018-2017-0-01633, ITRC(Information Technology Research Center) and No.2017-0-00613, Development of Content-oriented Delay Tolerant networking in Multi-access Edge Computing Environment)

References

1. (2013) Network Function Virtualization (NFV), Architecture framework, ETSI GS NFV 002 v 1.1.1
2. (2014) Network Function Virtualization (NFV), Management and Orchestration, ETSI GS NFV-MAN 001 v 1.1.1
3. (2017) Architecture enhancements for control and user plane separation of EPC nodes, TS 23.214 v15.0.0
4. (2017) Procedures for the 5G System, 3GPP TS 23.502 v1.3.0
5. (2017) System Architecture for the 5G System, 3GPP TS 23.501 v1.4.0
6. Baba H, Matsumoto M, Noritake K (2015) Lightweight virtualized evolved packet core architecture for future mobile communication. In: Proceedings of IEEE wireless communications and networking conference (WCNC). New Orleans, pp 1812–1816
7. Bagaa M, Taleb T, Ksentini A (2014) Service-aware network function placement for efficient traffic handling in carrier cloud. In: 2014 IEEE wireless communications and networking conference (WCNC). Istanbul, pp 2402–2407
8. Basta A, Kellerer W, Hoffmann M, Morper HJ, Hoffmann K (2014) Applying NFV and SDN to LTE mobile core gateways, the functions placement problem. In: Proceedings of the 4th workshop on all things cellular: operations, applications, and challenges. Chicago, pp 33–38
9. Basta A, Blenk A, Hoffmann K, Morper HJ, Hoffmann M, Kellerer W (2017) Towards a cost optimal design for a 5G mobile core network based on SDN and NFV. IEEE Trans Netw Serv Manag 14:1061–1075
10. Baumgartner A, Reddy VS, Bauschert T (2015) Mobile core network virtualization: a model for combined virtual core network function placement and topology optimization. In: Proceedings of the 2015 1st IEEE conference on network softwarization (NetSoft). London, pp 1–9
11. Chen M, Qian Y, Mao S, Tang W, Yang X (2016) Software-defined mobile networks security. Mob Netw Appl 21(5):729–743
12. Do TX, Nguyen VG, Kim Y (2016) SDN-based mobile packet core for multicast and broadcast services. Wireless Networks
13. Erel M, Teoman E, Özçevik Y, Seçinti G, Canberk B (2015) Scalability analysis and flow admission control in mininet-based sdn environment. In: 2015 IEEE conference on network function virtualization and software defined network (NFV-SDN). San Francisco, pp 18–19
14. Gurobi Optimization I (2017) Gurobi optimizer reference manual. http://www.gurobi.com
15. Kiess W, An A, Beker S (2015) Software-as-a-service for the virtualization of mobile network gateways. In: Proceedings of IEEE global communications conference (GLOBECOM 2015). San Diego, pp 1–6
16. Lange S, Gebert S, Zinner T, Tran-Gia P, Hock D, Jarschel M, Hoffmann M (2015) Heuristic approaches to the controller placement problem in large scale SDN networks. IEEE Trans Netw Serv Manag 12:4–17
17. Mavromatis I, Tassi A, Rigazzi G, Piechocki RJ, Nix A (2018) Multi-radio 5g architecture for connected and autonomous vehicles: application and design insights. EAI Endorsed Trans Indus Netw Intell Syst 4:13
18. NGMN P1 WS1 E2E Architecture Team (2016) Description of network slicing concept. Tech. rep., NGMN Alliance
19. Nguyen LD (2018) Resource allocation for energy efficiency in 5g wireless networks. EAI Endorsed Trans Indus Netw Intell Syst 5:14
20. Nguyen N, Duong TQ, Ngo HQ, Hadzi-Velkov Z, Shu L (2016) Secure 5g wireless communications: a joint relay selection and wireless power transfer approach. IEEE Access 4:3349–3359
21. Nguyen VG, Do TX, Kim Y (2016) SDN and virtualization-based LTE mobile network architectures: a comprehensive survey. Wirel Pers Commun 86(3):1401–1438
22. Nguyen VG, Brunstrom A, Grinnemo KJ, Taheri J (2017) SDN/NFV-based mobile packet core network architectures: a survey. IEEE Commun Surv Tutor 19(3):1567–1602
23. Obadia M, Bouet M, Rougier JL, Iannone L (2015) A greedy approach for minimizing SDN control overhead. In: Proceedings of the 2015 1st IEEE conference on network softwarization (NetSoft). London, pp 1–5
24. Open Networking Foundation (2014) SDN architecture. ONF TR-502
25. Paper CW (2017) Evolving the mobile core to being cloud native
26. Pentikousis K, Wang Y, Hu W (2013) Mobileflow: toward software-defined mobile networks. IEEE Commun Mag 51(7):44–53
27. Secinti G, Canberk B, Duong TQ, Shu L (2017) Software defined architecture for vanet: a testbed implementation with wireless access management. IEEE Commun Mag 55(7):135–141
28. Steffen G, David H, Thomas Z, Phuoc TG, Marco H, Michael J, Ernst-Dieter S, Banse R-PB (2014) Demonstrating the optimal placement of virtualized cellular network functions in case of large crowd events. In: Proceedings of the 2014 ACM conference on SIGCOMM. New York, pp 359–360
29. Taleb T, Ksentini A (2013) Gateway relocation avoidance-aware network function placement in carrier cloud. In: Proceedings of the 16th ACM international conference on modeling, analysis and simulation of wireless and mobile systems. Barcelona
30. Taleb T, Bagaa M, Ksentini A (2015) User mobility-aware virtual network function placement for virtual 5g network infrastructure. In: 2015 IEEE international conference on communications (ICC). London, pp 3879–3884
31. Vo N, Duong TQ, Guizani M, Kortun A (2018) 5g optimized caching and downlink resource sharing for smart cities. IEEE Access 6:31,457–31,468
32. Vo N, Duong TQ, Tuan HD, Kortun A (2018) Optimal video streaming in dense 5g networks with d2d communications. IEEE Access 6:209–223
33. de Weck O, Kim IY (2005) Adaptive weighted sum method for bi-objective optimization. Struct Multidiscip Optim 29(2):149–158
34. Xin J, Erran LL, Laurent V, Jennifer R (2013) SoftCell: scalable and flexible cellular core network architecture. In: Proceedings of the Ninth Acm conference on emerging networking experiments and technologies. Santa Barbara, pp 163–174