On **Dynamic Service Function Chain** Deployment and Readjustment

Junjie Liu, Wei Lu, Fen Zhou, Ping Lu, and Zuqing Zhu, Senior Member, IEEE

Outline

- Introduction
- Problem Description
- ILP Formulation
- Conlumn Generation Based Approach
- Performance Evaluation And Discussion

Introduction

Problems:

- service providers rely on middleboxes to realize the network functions
- The drawbacks of **dedicated hardware** results in difficulties in deployment and maintenance, prolonged time to market, and high expenses
- Cannot adapt to cloud computing and big data

Introduction

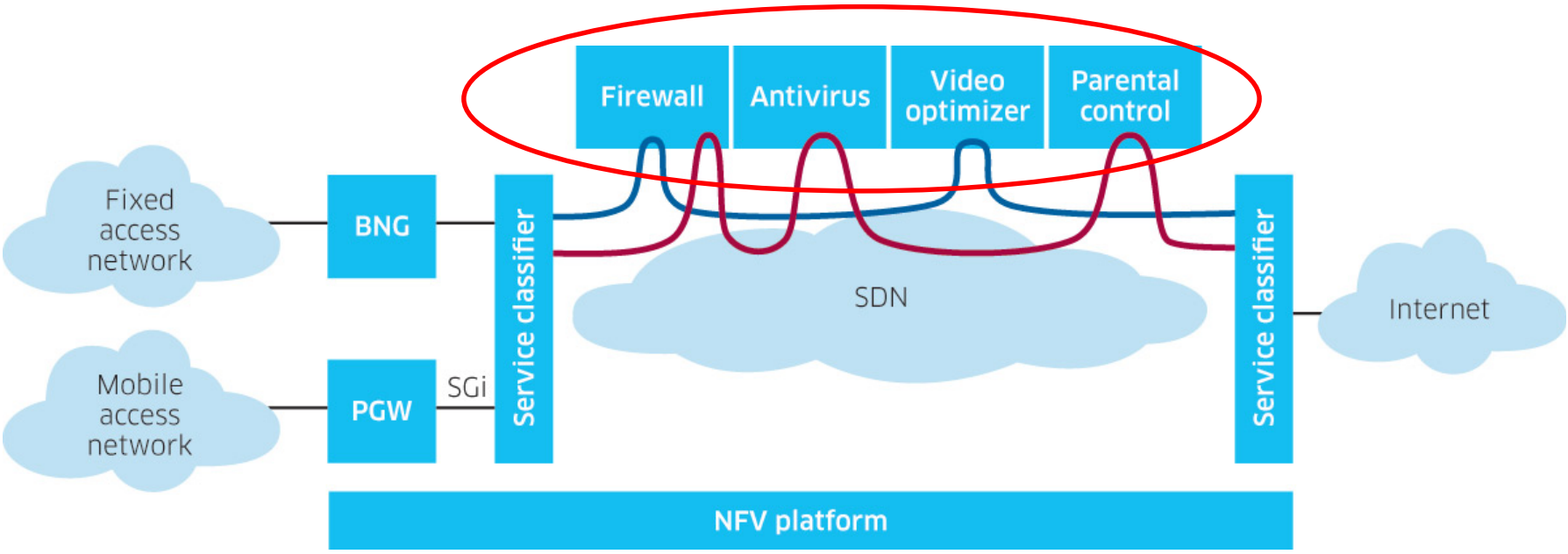
Network Function Virtualization(NFV):

- Dedicated hardware--> Software Defined Elements
- IT Resource Virtualization
- Processes traffic with *virtual network functions*(VNF)
- Don't worry hardware environment

Introduction

Service Function Chain(SFC):

- Network service with a series of connected VNFs



Challenge for networking control and management for the constraints on IT and Bandwidth resources

Introduction

Challenges:

- More and more mobile users are trying to access their services wherever and whenever.
- Consequently, SFCs need to be readjusted to ensure service continuity when the users are moving
- VNFs might be added in or removed from SFCs to adapt to their demands.

Introduction

Static SFC deployment:

- Make the paths of users and VNFs sub-optimal
- Unnecessary bandwidth consumption
- degrade user experience.
- cannot change their SFC patterns on-the-fly

Dynamic SFC deployment:

- Overhead from VNFs migrations is high
- Need to balance the tradeoff between resource consumption and operational overhead.

Introduction

Goals:

- Optimize VNFs **deployment** and **reassignment** to efficiently orchestrate SFCs in respond to **dynamic** user demands.

Introduction

Scenario Considerations:

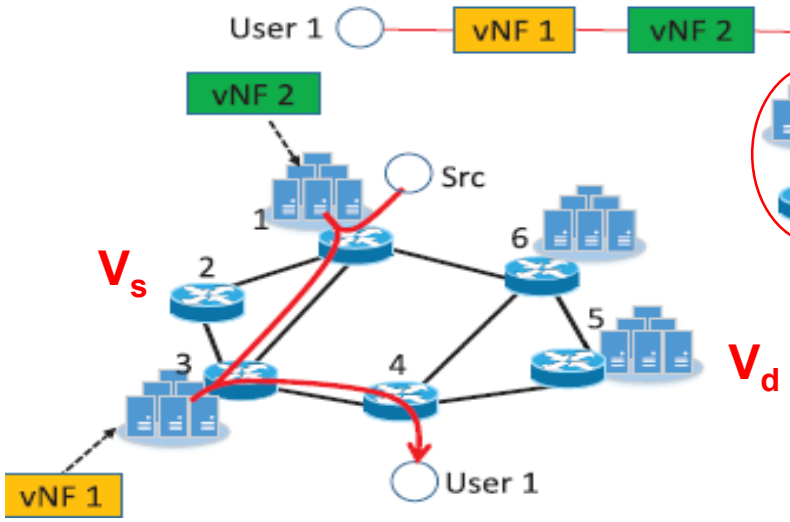
- VNF Deployment: IT and Bandwidth resource consumption
- VNF reassignment: the associated operational overhead.

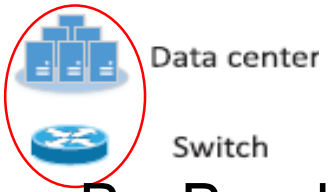
Introduction

Specific Methods:

- **Formulate** an integer linear programming (ILP) **model** to **maximize** the service provider's **profit** under the resource and operational overhead
- Design a column generation (CG) modelto approximate the performance of the ILP.

Problem Description





NetWork Model:

- G = (V, E)
- $V = V_d \cup V_s$

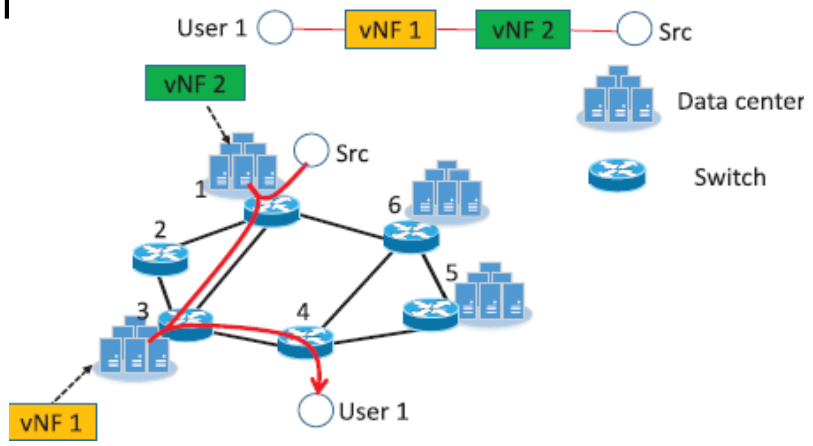
- B_e: Bandwidth Capacity
- C_v: IT Resource Capacity of a DC
- p: Set of NFV types instantiated in the DCs
- a p-th type VNF denoted its IT resource requirement as c_p can server np users at most.

- Switches: the access points for mobile users and forward data traffic in the network
- DCs contain the IT resources for VNF deployment
- Each DC is locally attached to a switch.

Problem Description

User i's service = $\langle v_i, \overrightarrow{g_i}, s_i, \beta_i, r_i \rangle$

- $v_i \in V_s$:Switch for its current access
- g_i Required SFC(Series of VNFs)
- s_i: source of traffic flow
- β_i: Bandwidth requirement on connection between two adjacent VNF
- r_i: profit for the service



Problem Description

Dynamic SFC Deployment and Readjustment

- $\Gamma = \Gamma_1 \cup \Gamma_2$
- Γ_1: The set of new users that are at the first time to access the network
- Γ_2: in-service users
- The overhead of the service provider is mainly from allocating new users to their required VNFs and migrating the services of in-service users to VNFs at new locations.
- Count how many VNFs the service provider operates for it, and use the total number as its operational overhead
- Maximize the service provider's profit = The total profit - the total deployment cost.

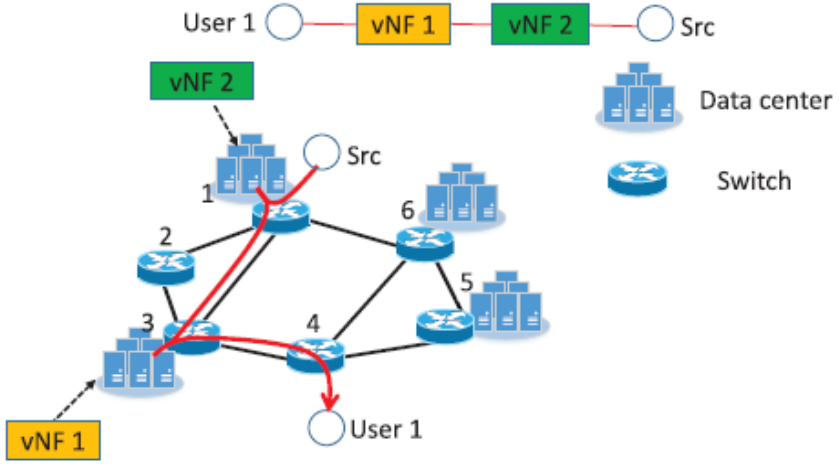
Problem Description

Dynamic SFC Deployment and Readjustment

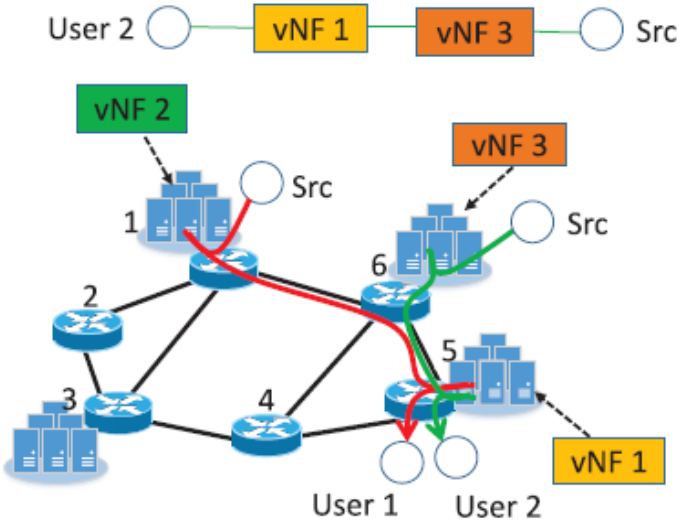
- To maximize the profit, the service provider needs to utilize both the IT and bandwidth resources properly, i.e., deploying vNFs adaptively in the DCs, and setting up routing paths intelligently to connect the vNFs for formulating the required SFCs for users.

Problem Description

- **Previous Service:** the in-service User 1 accesses the network from Switch 4 and takes an SFC as vNF 1→vNF 2. Hence, to set up the SFC for User 1, the service provider deploys a vNF 1 and a vNF 2 on DCs 3 and 1
- **Current Service:** A new User 2 joins at Switch 5 to request an SFC as vNF 1→vNF 3 and the in-service User 1 changes its access point to Switch 5 but its SFC stays unchanged. service provider decides to migrate the vNF 1 to DC 5 where it can be efficiently shared by Users 1 and 2, and deploys a vNF 3 on DC 6 for User 2.



(a) SFC provisioning scheme at previous service time.

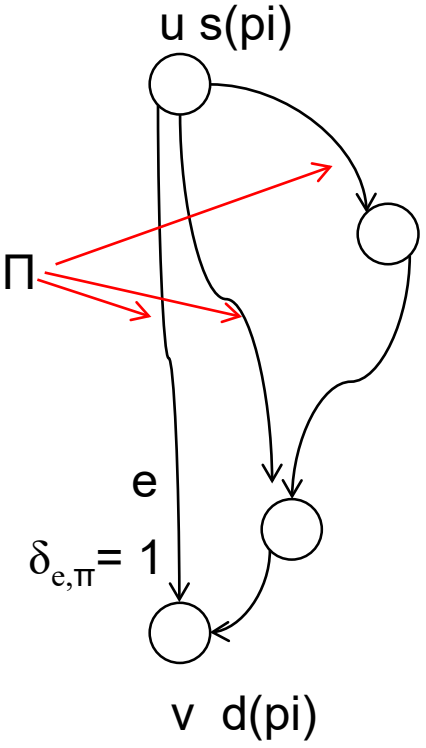


(b) SFC provisioning scheme at current service time.

ILP FORMULATION

Parameters:

- $G(V, E)$: Network topology, where $V = V_d \bigcup V_s$ and V_d and V_s are for the sets of DCs and switches, respectively.
- C_v: IT resource capacity of a DC $v \in V_d$.
- B_e: Bandwidth resource capacity of a link $e \in E$.
- Π: Set of pre-calculated paths in $G(V, E)$.
  - $\Pi_{u,v}$: Set of K shortest paths between u and v in $G(V, E)$, where $u, v \in V$ and $\Pi_{u,v} \subset \Pi$.
  - π: A path in Π, $s(\pi)$ and $d(\pi)$ represent the source and destination of π, respectively, and $len(\pi)$ denotes the hop-count of π.
  - $\delta_{e,\pi}$: A boolean constant to indicate whether a path π uses a link e. If yes, we have $\delta_{e,\pi} = 1$, and 0 otherwise.
- H_{max}: Maximum operational overhead that the service provider can take at each service time.
- $\mathcal{P}$: Set of supported vNF types in the network.
  - n_p: Maximum number of users that a p-th type vNF can serve ($p \in \mathcal{P}$).
  - c_p: IT resource consumption of a p-th type vNF.

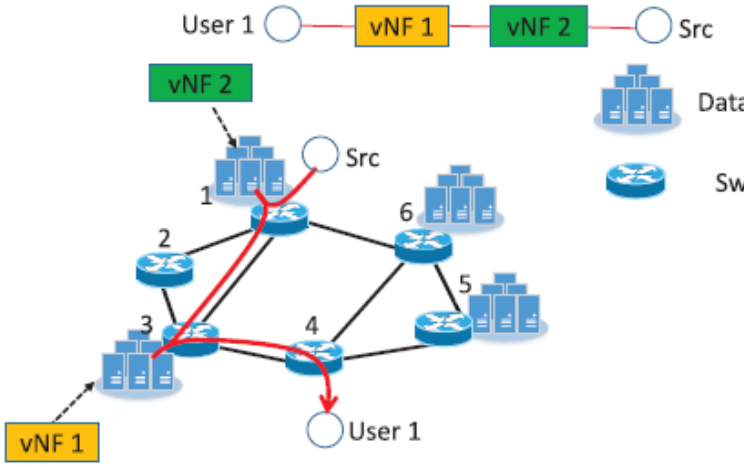


ILP FORMULATION

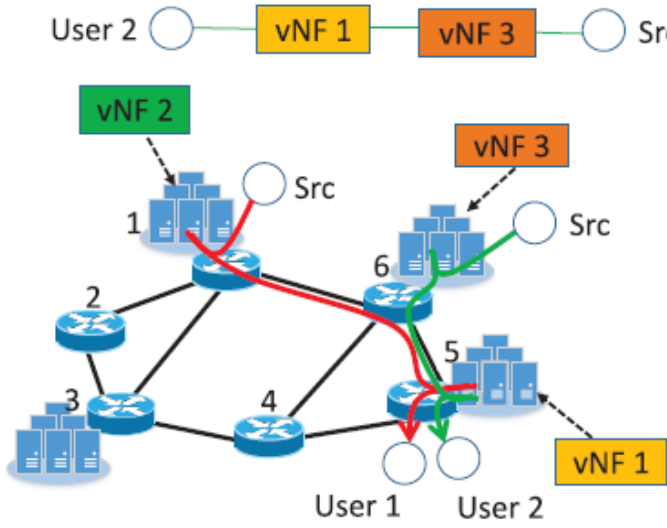
- Γ: Set of users to handle at the current service time, which includes both the new ones in Γ_1 and the in-service ones in Γ_2. We use $|\Gamma|$ to denote the total number of users.
- $\langle v_i, \overrightarrow{g_i}, s_i, \beta_i, r_i \rangle$: A 5-tuple to represent the service request of a user i at current service time.
  - v_i: Access switch of a user i, $v_i \in V_s$.
  - $\overrightarrow{g_i}$: Required SFC of a user i, which is a vector that consists of a sequence of specific vNFs.
    - L_i: Number of vNFs in the SFC of a user i. For convenience, we treat the access switch of a user as a dummy vNF that consumes 0 IT resource, *i.e.*, the first vNF in each user's SFC is its access switch while the other ones are real vNFs. Hence, we have $L_i = |\overrightarrow{g_i}| + 1$.
    - $g_{i,j,p}$: A boolean constant that equals 1 if the j-th vNF on the SFC of a user i is a p-th type vNF, and 0 otherwise. Since the first vNF is the user's access switch, we have $g_{i,1,p} = 0, \ \forall i, p$. We also assume that there are no duplicated vNFs in a user's SFC, and thus if $j_1 \neq j_2$, we have $g_{i,j_1,p} + g_{i,j_2,p} \leqslant 1, \ \forall i, p$.

- s_i: The source of the required traffic flow,[2] $s_i \in V_d$.
- β_i: Bandwidth requirement on a connection between two adjacent vNFs in the SFC of a user i.
- r_i: Profit that the service provider gains after provisioning the SFC of a user i.

- $\langle d_{i,j,p}, a_{i,j,v} \rangle$: A 2-tuple to represent the SFC provisioning scheme of an in-service user i before current service time.
  - $d_{i,j,p}$: A boolean constant to represent the required SFC of an in-service user i before current service time. It equals 1 if the j-th vNF on the SFC of a user i is a p-th type vNF, and 0 otherwise. It corresponds to parameter $g_{i,j,p}$ described above for the current service time. It should be noted that $d_{i,j,p}$ might not equal $g_{i,j,p}$ since vNFs might be added in or deleted from an SFC due to the dynamic operation.
  - $a_{i,j,v}$: A boolean constant to indicate that for an in-service user i, whether the j-th vNF in its SFC was deployed on node v before current service time. If yes, we have $a_{i,j,v} = 1$, and 0 otherwise. This parameter corresponds to variable $z_{i,j,v}$ below for current service time.
  - $f_{i,v,p}$: For convenience, we integrate $d_{i,j,p}$ and $a_{i,j,v}$ to obtain another parameter $f_{i,v,p}$ to describe whether an in-service user i used the p-th vNF on DC v before the current service time, *i.e.*, $f_{i,v,p} = \sum_j a_{i,j,v} \cdot d_{i,j,p}$.



(a) SFC provisioning scheme at previous service ti



(b) SFC provisioning scheme at current service tin

Variables:

- x_i: A boolean variable that equals 1 if the SFC of a user i is served at current service time, and 0 otherwise.
- $y_{v,p}$: An integer variable to indicate the number of p-th type vNFs that are deployed on a DC $v \in V_d$.
- $z_{i,j,v}$: A boolean variable that equals 1 if the j-th vNF of a user i is deployed on a DC $v \in V_d$, and 0 otherwise.
- $w_{i,j,\pi}$: A boolean variable that equals 1 if for a user i, the connection from its j-th vNF to $(j+1)$-th vNF (*i.e.*, $j \in [1, L_i - 1]$) uses the path π, and 0 otherwise.

Constraints:

1) *vNF Assignment and Readjustment Constraints:*

$$\sum_{v \in V_d} z_{i,j,v} = x_i, \quad \forall j \in [2, L_i], \ \forall i \in \Gamma. \tag{1}$$

Eq. (1) ensures that if a user i is served at the current service time, the required vNFs in its SFC are all realized on the DCs.

$$z_{i,1,v_i} = x_i, \quad \forall i \in \Gamma, \tag{2}$$

$$z_{i,L_i,s_i} = x_i, \quad \forall i \in \Gamma. \tag{3}$$

Eqs. (2)-(3) ensure that if a user i is served at the current service time, its first vNF is a dummy one that is allocated on its access switch while its last vNF represents the source of the traffic flow s_i.

$$\sum_i \sum_j g_{i,j,p} \cdot z_{i,j,v} \leq n_p \cdot y_{v,p}, \quad \forall v \in V_d, \ \forall p \in \mathcal{P}. \tag{4}$$

Eq. (4) ensures that the number of users that use any type of vNFs deployed in a DC v cannot exceed their total capacity.

$$\sum_p c_p \cdot y_{v,p} \leq C_v, \quad \forall v \in V_d. \tag{5}$$

Eq. (5) ensures that the vNFs deployed on a DC v cannot use more IT resources than its IT resource capacity.

$$\sum_{v \in V_d} \left\{ \sum_{i \in \Gamma_1} \sum_{j=2}^{L_i} z_{i,j,v} + \sum_{i \in \Gamma_2} \sum_{j=2}^{L_i} \left[\sum_p (1 - f_{i,v,p}) \cdot g_{i,j,p} \right] \cdot z_{i,j,v} \right\} \leq H_{max}. \tag{6}$$

Eqs. (2)-(3) ensure that if a user i is served at the current service time, its first vNF is a dummy one that is allocated on its access switch while its last vNF represents the source of the traffic flow s_i.

$$\sum_{i}\sum_{j} g_{i,j,p} \cdot z_{i,j,v} \leq n_p \cdot y_{v,p}, \quad \forall v \in V_d, \ \forall p \in \mathcal{P}. \tag{4}$$

Eq. (4) ensures that the number of users that use any type of vNFs deployed in a DC v cannot exceed their total capacity.

$$\sum_{p} c_p \cdot y_{v,p} \leq C_v, \quad \forall v \in V_d. \tag{5}$$

Eq. (5) ensures that the vNFs deployed on a DC v cannot use more IT resources than its IT resource capacity.

$$\sum_{v \in V_d} \left\{ \sum_{i \in \Gamma_1} \sum_{j=2}^{L_i} z_{i,j,v} + \sum_{i \in \Gamma_2} \sum_{j=2}^{L_i} \left[\sum_{p} (1 - f_{i,v,p}) \cdot g_{i,j,p} \right] \cdot z_{i,j,v} \right\} \leq H_{max}. \tag{6}$$

Eq. (6) ensures that the service provider's operational overhead at the current service time would not exceed the preset upper-limit H_{max}. Here, the first term corresponds to the overhead to handle the new users while the second one is for the overhead to process the in-service ones. We use $\sum_p (1 - f_{i,v,p}) \cdot g_{i,j,p}$ to determine whether the j-th vNF on the SFC of a user i has been readjusted or not at the current service time.

2) *Routing and Bandwidth Allocation Constraints:*

$$\sum_{i}\sum_{j}\sum_{\pi\in\Pi}\delta_{e,\pi}\cdot\beta_i\cdot w_{i,j,\pi}\leq B_e,\quad \forall e\in E. \tag{7}$$

Eq. (7) ensures that the total bandwidth consumption on each link $e \in E$ would not exceed its bandwidth capacity.

$$\sum_{\pi} w_{i,j,\pi} = x_i,\quad \forall j\in[1,L_i-1],\ \forall i\in\Gamma. \tag{8}$$

Eq. (8) ensures that for each served user i at the current service time, each of its connections between two adjacent vNFs takes one and only one path in the network.

$$z_{i,j,v} = \sum_{s(\pi)=v} w_{i,j,\pi},\quad \forall i\in\Gamma,\ \forall v\in V,\ \forall j\in[1,L_i-1], \tag{9}$$

$$z_{i,(j+1),v} = \sum_{d(\pi)=v} w_{i,j,\pi},\quad \forall i\in\Gamma,\ \forall v\in V,\ \forall j\in[1,L_i-1]. \tag{10}$$

Eqs. (9) and (10) ensure that if a path π is selected to carry the connection between the j-th vNF and the $(j+1)$-th vNF of a user i, the j-th and $(j+1)$-th vNFs are deployed on $s(\pi)$ and $d(\pi)$, respectively.

Objective: The optimization objective is to maximize the service provider's profit, which is the total profit from the served requests minus the total deployment cost. The deployment cost can be described as follows.

$$\mathbb{C} = \rho_1 \cdot \sum_{v \in V_d} \sum_{p} c_p \cdot y_{v,p} + \rho_2 \cdot \sum_{i} \sum_{j=1}^{L_i - 1} \sum_{\pi} \beta_i \cdot len(\pi) \cdot w_{i,j,\pi}, \tag{11}$$

where ρ_1 and ρ_2 are the positive coefficients to quantify the unit costs of IT and bandwidth resource consumptions, respectively. The profit can be calculated as

$$\mathbb{R} = \sum_{i} r_i \cdot x_i, \tag{12}$$

and we finalize the optimization objective as

$$\textit{Maximize} \quad \mathbb{R} - \mathbb{C}. \tag{13}$$

ILP FORMULATION

NP-hard problem

- Be = +∞, ρ2 = 0 bandwidth constraint is ignored
- Secondly, we apply the restriction that both the number of supported vNF types in the network and the number of vNFs in the SFC of each user are 1. Lastly,
- we apply the restriction that Hmax = +∞.

CG Approach

- Decomposing the original problem into a **master** problem and a **pricing** problem
- Solving them in iterations to obtain the **near-optimal solution** to the original problem

Algorithm 1: General Procedure of CG-Based Approach

```
1  define the notes to represent a column c;
2  formulate the ILP model for master problem (ILP-MP);
3  formulate the ILP model for pricing problem (ILP-PP);
4  C = ∅;
5  generate a column c for ILP-MP that all the parameters
   in the notes are 0;
6  insert c into C to obtain the initial solution;
7  use C to construct the LP relaxation of ILP-MP;
8  while TRUE do
9      solve the LP relaxation of ILP-MP to get the values
       of primal and dual variables;
10     construct ILP-PP with the results from the LP
       relaxation of ILP-MP;
11     solve ILP-PP to get its optimization objective ℚ;
12     if ℚ ≥ 0 then
13         break;
14     end
15     generate a new column c with the results of ILP-PP;
16     insert c into C;
17     use C to update the LP relaxation of ILP-MP;
18 end
19 use C to construct ILP-MP;
20 solve ILP-MP to obtain the final solution;
```

Initialize

Relaxation?
How to generate column?

CG Approach

- a solution to our problem is just a combination of certain feasible SFC provisioning schemes of the users
- denote a feasible SFC provisioning scheme of a user as a column c
- $Q \geq 0$ means that the objective of the original problem cannot be reduced anymore

Master problem

Variables:

- λ_c: A boolean variable to indicate whether a column $c \in C$ is selected. If yes, we have $\lambda_c = 1$, and 0 otherwise.
- $y_{v,p}$: An integer variable to indicate the number of p-th type vNFs that are deployed on a DC $v \in V_d$.

Objective: For convenience, we add a minus sign to the objective in Eq. (13) to obtain the equivalent form of minimization. As we only consider the provisioning schemes included in C in the master problem, the objective of the dynamic SFC deployment and readjustment is transformed into

$$\begin{aligned} \textit{Minimize} \quad & \sum_{c \in C} \lambda_c \cdot \left(\rho_2 \cdot \sum_{e} b_{c,e} - \sum_{i} r_i \cdot m_{i,c} \right) \\ & + \rho_1 \cdot \sum_{v \in V_d} \sum_{p} c_p \cdot y_{v,p}. \end{aligned} \tag{14}$$

Master problem

Constraints: The constraints get changed as follows. Note that, we also list the corresponding dual variables here in "()", which indicate the reduced cost on the objective in Eq. (14). For these inequations with "$\leq$", we use negative dual variables to ensure that the values of the dual variables are not smaller than 0.

$$\sum_{c} h_c \cdot \lambda_c \leq H_{max}, (-\varphi), \tag{15}$$

$$\sum_{p} c_p \cdot y_{v,p} \leq C_v, \quad \forall v \in V_d, \quad (-\tau_v), \tag{16}$$

$$\sum b_{c,e} \cdot \lambda_c \leq B_e, \quad \forall e \in E, \quad (-\alpha_e). \tag{17}$$

Notes:

- $m_{i,c}$: If c is a provisioning scheme of user i, we have $m_{i,c} = 1$ and 0 otherwise.
- $a_{c,v,p}$: If c allocates a p-th type vNF on DC v, we have $a_{c,v,p} = 1$ and 0 otherwise.
- h_c: The operational overhead that the service provider would take to serve the user i that is selected in c.
- $b_{c,e}$: The bandwidth consumption on each used link e for the provisioning scheme in c.

Eqs. (15)-(17) ensure that the constraints on operational overhead, IT resources and bandwidth resources are satisfied.

$$\sum_{c} m_{i,c} \cdot \lambda_c \leq 1, \quad \forall i \in \Gamma, (-\gamma_i), \tag{18}$$

$$\sum_{c} \lambda_c \leq |\Gamma|, (-\eta). \tag{19}$$

Eqs. (18), (19) ensure that for each user i, at most one column can be selected.

$$\sum_{c} a_{c,v,p} \cdot \lambda_c \leq n_p \cdot y_{v,p}, \quad \forall v \in V_d, \ \forall p \in \mathcal{P}, \left(-\chi_{v,p}\right). \tag{20}$$

Eq. (20) ensures that for each p-th type vNF on a DC v, the number of users that use it cannot exceed its capacity.

Pricing problem

Variables:

- $x_i, z_{i,j,v}, w_{i,j,\pi}$: Variables that have the same definitions as those in Section IV.

The relations between these variables and the notes $m_{i,c}, a_{c,v,p}, h_c$ and $b_{c,e}$ defined for a specific column c are as follows.

$$m_{i,c} = x_i, \quad \forall i \in \Gamma, \tag{21}$$

which is because in each iteration, we only consider one column that corresponds to the provisioning scheme of a user.

$$h_c = \sum_{i\in\Gamma_1} \sum_{j=2}^{L_i} \sum_{v\in V_d} z_{i,j,v} + \sum_{i\in\Gamma_2} \sum_{j=2}^{L_i} \sum_{v\in V_d} \left[\sum_{p} (1 - f_{i,v,p}) \cdot g_{i,j,p} \right] \cdot z_{i,j,v}, \tag{22}$$

which is because h_c denotes the total operational overhead.

$$a_{c,v,p} = \sum_{i} \sum_{j} g_{i,j,p} \cdot z_{i,j,v}, \quad \forall v \in V_d, \ \forall p \in \mathcal{P}, \tag{23}$$

where the item $g_{i,j,p} \cdot z_{i,j,v}$ indicates whether the j-th vNF on the SFC of a user i is assigned to a p-th type vNF on the DC v. Hence, with a summation, we can determine whether a p-th type vNF on the DC v is used in the column c.

$$b_{c,e} = \sum_{i} \sum_{j} \sum_{\pi} \beta_i \cdot \delta_{e,\pi} \cdot w_{i,j,\pi}, \quad \forall e \in E, \tag{24}$$

which gets the bandwidth usage on link e. Note that, based on the relation between the primal and dual problems [42], the reduced cost of λ_c has the expression as

$$\sum_{i} m_{i,c} \cdot (\gamma_i - r_i) + \sum_{e} b_{c,e} \cdot (\rho_2 + \alpha_e) + \sum_{v \in V_d} \sum_{p} \chi_{v,p} \cdot a_{c,v,p} + \varphi \cdot h_c + \eta. \tag{25}$$

Objective: Then, we put Eqs. (21)-(24) into Eq. (25) and get the objective of the pricing problem as

$$\begin{aligned} \textit{Minimize} \quad \mathbb{Q} = & \sum_{i,j,\pi} \zeta_{i,j,\pi} \cdot w_{i,j,\pi} + \sum_{i \in \Gamma_1, j, v} \tilde{\zeta}_{i,j,v} \cdot z_{i,j,v} \\ & + \sum_{i \in \Gamma_2, j, v} \hat{\zeta}_{i,j,v} \cdot z_{i,j,v} + \sum_{i} \zeta_i \cdot x_i + \eta, \end{aligned} \tag{26}$$

where we use the following notations to simplify the objective:

$$\begin{cases} \zeta_{i,j,\pi} = \beta_i \cdot \sum_e (\rho_2 + \alpha_e) \cdot \delta_{e,\pi}, \\ \tilde{\zeta}_{i,j,v} = \varphi + \sum_p \chi_{v,p} \cdot g_{i,j,p}, \\ \hat{\zeta}_{i,j,v} = \sum_p \left[\left(1 - f_{i,v,p}\right) \cdot \varphi + \chi_{v,p} \right] \cdot g_{i,j,p}, \\ \zeta_i = \gamma_i - r_i. \end{cases} \tag{27}$$

Constraints: The ILP-PP inherits the constraints in Eqs. (1)-(3) and (8)-(10) defined in Section IV. Moreover, we introduce the following constraints to expedite the solving of the ILP-PP.

$$\sum_{i} x_i = 1. \tag{28}$$

Eq. (28) ensures that the pricing problem only consider the service provisioning of one user.

$$\begin{aligned} \sum_{i}\sum_{j}\left(\sum_{p} c_p \cdot g_{i,j,p}\right) \cdot z_{i,j,v} \leq C_v, \quad \forall u \in V_d, \\ \sum_{i}\sum_{j}\sum_{\pi} \delta_{e,\pi} \cdot \beta_i \cdot w_{i,j,\pi} \leq B_e, \quad \forall e \in E. \end{aligned} \tag{29}$$

Eq. (29) ensures that in each obtained column, the capacity constraints on IT and bandwidth resources are satisfied, *i.e.,* the obtained column is a feasible solution. Hence, the ILP-MP can be constructed correctly in *Line* 19 of *Algorithm* 1.

Algorithm 2: Heuristic to Solve ILP-PP

assign the weight of each link $e \in E$ as $\rho_2 + \alpha_e$ according to Eq. (27);
calculate the weighted shortest paths between all the node pairs in the network;
for *each user* $i \in \Gamma$ **do**
    for *each vNF* $j \in [2, L_i]$ **do**
        use the greedy strategy to find the DC v for this vNF such that $\mathbb{Q}$ in Eq. (26) is minimized except the L_i-th vNF which is fixed;
        connect the $(j-1)$-th and j-th vNFs of user i with the weighted shortest path between them;
        calculate the incremental cost as $Q_{i,j}$;
    end
    get the overall cost of user i as $Q_i = \sum_j Q_{i,j}$;
end
$\mathbb{Q} = \min_{i \in \Gamma}(Q_i)$;

Relaxation?
How to generate column?

Algorithm 3: Sub-Procedure to Accelerate CG

construct pricing problem with the results from the LP relaxation of ILP-MP;
solve the pricing problem with *Algorithm* 2 to get $\mathbb{Q}$;
if $\mathbb{Q} \geq 0$ **then**
    construct ILP-PP with the results from the LP relaxation of ILP-MP;
    solve ILP-PP to get its optimization objective $\mathbb{Q}$;
    if $\mathbb{Q} \geq 0$ **then**
        break;
    end
end
generate a new column c with results of pricing problem;

PERFORMANCE EVALUATION AND DISCUSSION

- small **six-node one** as shown in Fig. 1 and a practical NSFNET topology
- Use the **5-shortest paths** between two nodes in the NSFNET topology to replace the total paths when using the proposed-CG model.
- assume that the DCs are all light-weighted ones and they are **randomly distributed** in the topologies, which means not all the switches have local DCs.

PERFORMANCE EVALUATION AND DISCUSSION

- The DCs' IT resource capacities are
- **uniformly distributed** within [30, 35] units
- Bandwidth capacities of each physical links are **randomly selected** within [25, 30] units.
- The upper-limit on the service provider's operational overhead (i.e., Hmax) is set within [30, 60] for each service time

PERFORMANCE EVALUATION AND DISCUSSION

- treat the costs of IT and bandwidth resources **equally** in the optimization, and thus the simulations use **ρ1 = ρ2 = 1** in the objective.
- The network with the six-node topology can support **4 vNF** types, while the one with the NSFNET topology can support **5 vNF** types.

PERFORMANCE EVALUATION AND DISCUSSION

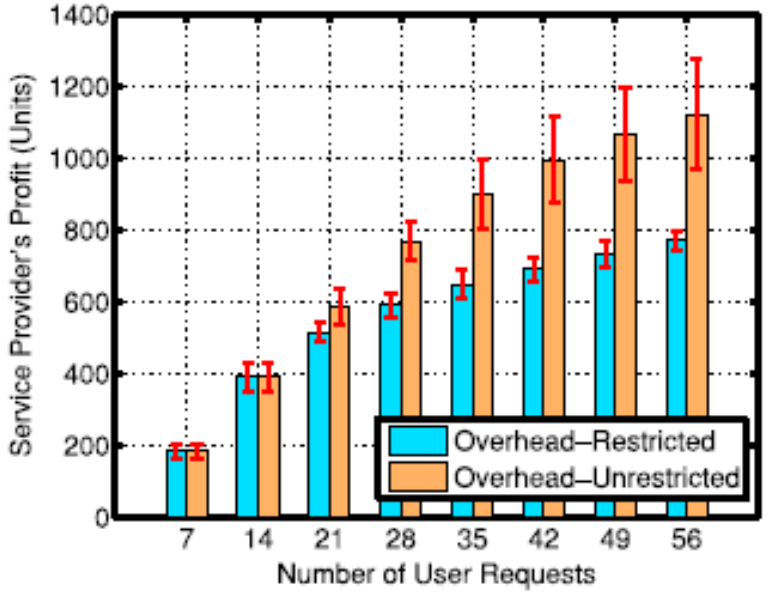
- For each vNF type, the IT resource consumption of a vNF is within **[5, 8]** units and each can serve **3 to 5** users at most.
- The number of vNFs requested on the SFC by each user is assumed to be **uniformly distributed** within [**2, 4**] and [**2, 5**] for the six-node topology and the NSFNET topology, respectively.

PERFORMANCE EVALUATION AND DISCUSSION

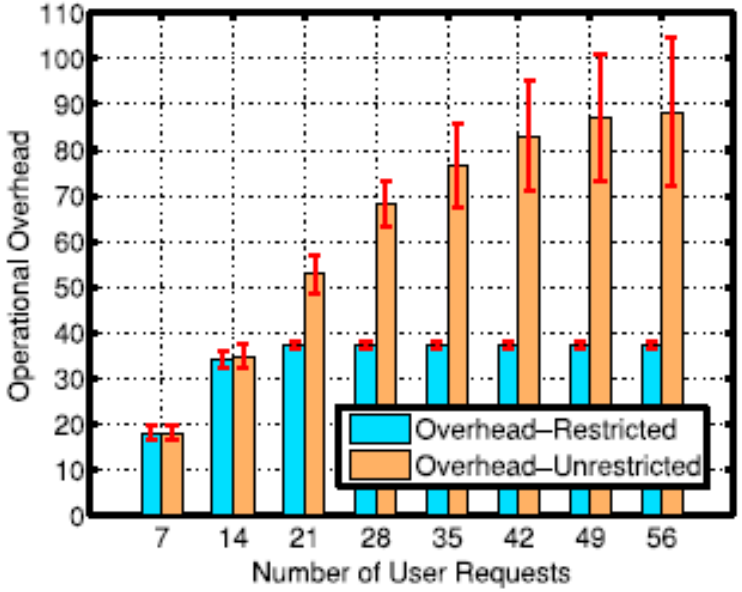
- For each vNF type, the IT resource consumption of a vNF is within **[5, 8]** units and each can serve **3 to 5** users at most.
- The number of vNFs requested on the SFC by each user is assumed to be **uniformly distributed** within [**2, 4**] and [**2, 5**] for the six-node topology and the NSFNET topology, respectively.

PERFORMANCE EVALUATION AND DISCUSSION

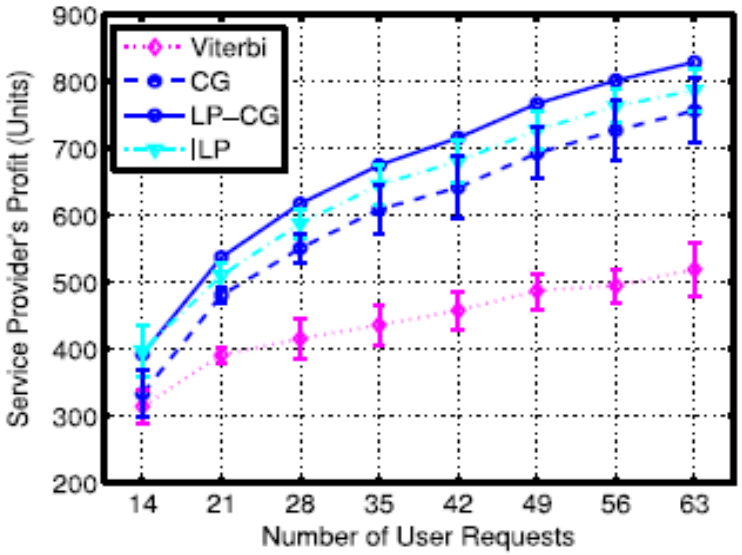
- the bandwidth requirement on links is within [**3, 5**] units.
- To obtain sufficient statistical accuracy, we obtain each data point by **averaging** the results from **20** independent simulations.



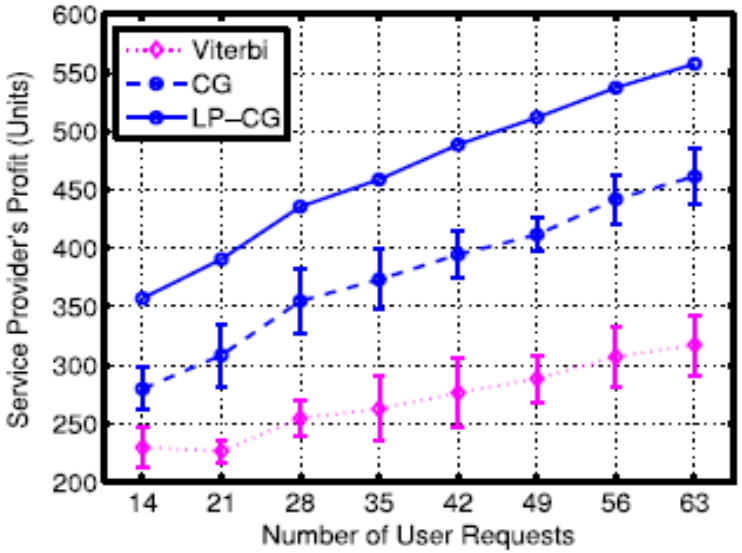
(a) Service Provider's Profit



(b) Operational Overhead



(a) Six-node topology



(b) NSFNET topology

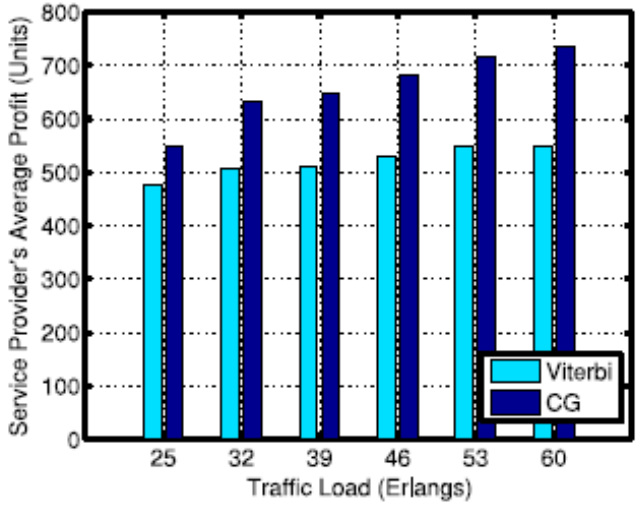
ig. 3. Results on service provider's profit in one-time operation.

TABLE I
COMPUTATION TIME PER REQUEST WITH SIX-NODE TOPOLOGY (SECONDS)

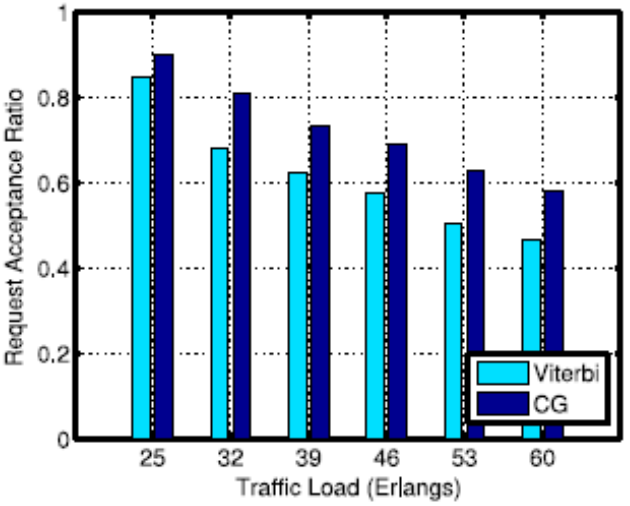
# of User Requests	Viterbi	CG	ILP
14	0.0068	0.15	0.071
21	0.0073	0.18	0.12
28	0.0079	0.20	0.27
35	0.0087	0.24	0.37
42	0.090	0.28	0.92
49	0.093	0.30	1.45
56	0.011	0.34	2.86
63	0.015	0.39	5.52

TABLE II
COMPUTATION TIME PER REQUEST WITH NSFNET TOPOLOGY (SECONDS)

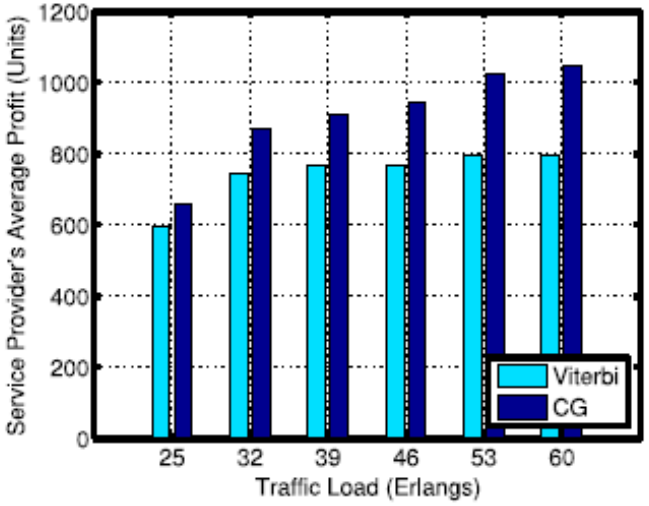
# of User Requests	Viterbi	CG
14	0.053	0.51
21	0.055	1.35
28	0.056	1.58
35	0.058	2.05
42	0.063	2.52
49	0.072	3.07
56	0.077	4.23
63	0.084	6.28



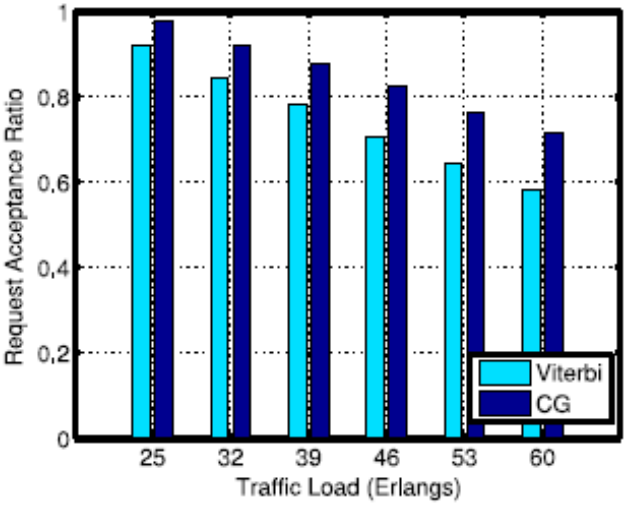
(a) Low overhead scenario



(a) Low overhead scenario



(b) High overhead scenario



(b) High overhead scenario

TABLE III
AVERAGE PROFIT RATIO BETWEEN CG AND VITERBI

Traffic Load (Erlangs)	25	32	39	46	53	60
Low Overhead	1.15	1.25	1.27	1.28	1.31	1.34
High Overhead	1.10	1.17	1.19	1.23	1.28	1.32

Conclusion

- formulated a path-based ILP model to solve the problem exactly.
- To reduce the time complexity, we designed a CG model, developed an approximation algorithm based on it and proposed an effective heuristic to further accelerate the problem-solving.
- CG-based approach significantly outperformed the benchmark algorithm in terms of the service provider’s profit and request acceptance ratio.

Thank You!