On Incentive-Driven VNF Service Chaining in Inter-Datacenter Elastic Optical Networks: A Hierarchical Game-Theoretic Mechanism

On Incentive-Driven VNF Service Chaining in Inter-Datacenter Elastic Optical Networks: A Hierarchical Game-Theoretic Mechanism

Xiaoliang Chen, *Member, IEEE*, Zuqing Zhu, *Senior Member, IEEE*, Roberto Proietti, S. J. Ben Yoo, *Fellow, IEEE, Fellow, OSA*

***Abstract*—In this paper, we propose an incentive-driven virtual network function service chaining (VNF-SC) framework for optimizing the cross-stratum resource provisioning in multi-broker orchestrated inter-datacenter elastic optical networks (IDC-EONs). The proposed framework employs a non-cooperative hierarchical game-theoretic mechanism, where the resource brokers and the VNF-SC users play the leader and the follower games, respectively. In the leader game, the brokers calculate VNF-SC service schemes for users and compete for the provisioning tasks. While in the follower game, the users compete for VNF-SC services for jointly optimizing the resource cost and the received quality-of-service. We first elaborate on the modeling of the follower game, discuss the existence of Nash equilibrium and propose a mixed-strategy gaming approach enabled by an auxiliary graph based algorithm to facilitate users selecting the most appropriate service schemes. Then, under the assumption that the brokers are aware of the principle of the follower game, we present the model for the leader game and develop a time-efficient heuristic algorithm for brokers to compete for the provisioning tasks. Simulations show that the proposed incentive-driven VNF-SC framework significantly improves the network throughput (*i.e.*, $> 4.8\times$ blocking reduction) while assisting users and brokers in achieving higher utilities compared with existing solutions.**

***Index Terms*—Virtual network function service chaining (VNF-SC), Inter-datacenter elastic optical networks (IDC-EONs), Multi-broker, Hierarchical gaming.**

I. Introduction

THE emerging of network function virtualization (NFV) has been renovating the ways of service provisioning in telecom and datacom networks [1, 2]. Specifically, NFV replaces proprietary hardware implementations with virtual network functions (VNFs) built with commodity hardware to realize programmable and application-aware network service (*e.g.*, firewall and deep packet inspection) provisioning. As one of the most important use cases of NFV, VNF service chaining (VNF-SC) makes users' traffic be steered by sequences of VNFs instantiated in datacenters (DCs) to meet heterogeneous service requirements [3, 4]. Therefore, the key problem of VNF-SC is how to efficiently form service function chains with cross-stratum resource (*i.e.*, bandwidth and IT resources) optimization. This is especially true in the context of inter-DC elastic optical networks (IDC-EONs) with high transmission capacity and agile spectrum allocation capability [5].

Previously, there had been a number of works focusing on addressing the problem of VNF placement in both packet [6–18] and optical networks [19–24]. However, they all assumed a single decision maker on top of the network and DC controllers to orchestrate the allocation of cross-stratum resources. Note that, inter-DC networks and DCs are usually managed by different administrative entities, making the aforementioned assumption unrealistic since it violates the well-established principle for autonomous systems (AS's). Meanwhile, as the existing schemes all aimed to realize joint resource optimizations for high network throughput, incentives from users were not considered. For example, some of the VNF-SCs with low latency requirements may be forced to use long transmission paths due to load-balancing. On the other hand, multi-broker based architecture that enables multiple resource brokers residing in the management plane has been proven to 1) be cost-efficient in facilitating service provisioning in multi-AS networks and 2) favors both infrastructure operators and users [25, 26]. In particular, each operator can subscribe to multiple brokers for resource coordination services to avoid being dictated by a single decision-maker while users can receive multiple service schemes calculated by brokers to get better services.

In this work, we take advantage of the multi-broker based network orchestration paradigm and propose an incentive-driven VNF-SC framework for IDC-EONs. We first detail the architecture of multi-broker based IDC-EON and the principle of incentive-driven VNF-SC provisioning. Then, the problem is formally modeled as a non-cooperative hierarchical game, where the brokers play the leader game to calculate VNF-SC service schemes for the users and compete for the provisioning tasks, while after being provided the service schemes, the users subsequently play the follower game to compete for spectrum and IT resources. Specifically, we assume that the utility of each user is related to the resource consumption cost and the end-to-end latency of its service function chain, and design a mixed-strategy game-theoretic approach for the users to decide the most appropriate service schemes to use. An auxiliary graph based algorithm is also proposed for calculating approximated equilibrium solutions for the users. Under the assumption that the brokers are aware of the information of the follower game, we present the model

X. Chen, R. Proietti and S. J. B. Yoo are with the Department of Electrical and Computer Engineering, University of California, Davis, Davis, CA 95616, USA (Email: xlichen@ucdavis.edu, sbyoo@ucdavis.edu).

Z. Zhu is with the School of Information Science and Technology, University of Science and Technology of China, Hefei, Anhui 230027, P. R. China (Email: zqzhu@ieee.org).

Manuscript received Feb. 8, 2018.

for the leader game and design a time-efficient heuristic to enable the brokers determining the best provisioning strategies. Numerical simulations indicate that comparing with the baseline algorithms, the proposed incentive-driven VNF-SC provisioning framework can improve the network throughput while facilitating higher user and broker utilities.

The rest of this paper is organized as follows. In Section II, we briefly review the related works on NFV. In Section III, we show the proposed incentive-driven VNF-SC provisioning framework in multi-broker based IDC-EONs. In Sections IV and V, we detail the designs for the follower and leader games, respectively. Simulation results are presented in Section VI. Finally, in Section VII, we conclude the paper.

II. Related Work

NFV has attracted intensive interests from both academia and industry since it was proposed. The overviews of NFV can be found in [2, 27], where the authors explained its requirements and architectural framework, discussed several practical use cases and presented surveys of the state-of-the-art and future directions of this area. The authors of [28, 29] studied in detail the management and orchestration architecture for NFV leveraging the software-defined networking (SDN) technology. One of the most important research problems of NFV is to investigate how to realize cost-effective VNF placement [7–18]. In [7], Cohen *et al.* proposed several near optimal approximation algorithms to address this problem without considering the resource constraints. The authors of [8] introduced an orchestrator-based architecture supported by a monitoring system for ensuring automatic VNF placement in DCs. However, they only used a simple algorithm for load-balancing the utilization of IT resources while the optimization of bandwidth allocations was not addressed. Moens *et al.* studied the problem of VNF placement for a hybrid scenario where dedicated physical hardware and virtualized service instances coexist in [9]. In [10, 11], VNF-SC with joint bandwidth and IT resource optimization in packet domains was investigated and mixed integer programming models were designed for solving the offline network planning problem. To cope with large-scale online VNF-SC, a number of time-efficient heuristic algorithms have been developed [12–16]. In [17], Liu *et al.* exploited the benefit of dynamic in-service VNF-SC readjustment, for which an integer linear programming model and a column generation based heuristic algorithm were proposed. The authors of [18] further extended the design of routing and VNF placement to support multicast applications.

It is known that optical technologies can facilitate high-capacity and energy-efficient data transmission which is especially beneficial for building inter-DC connections. The problem of VNF placement in wavelength-switched DC networks was first studied in [5]. Based on the same architecture, Xia *et al.* designed a binary integer programming model as well as an alternative efficient heuristic algorithm to minimize the usage of optical-electronic-optical (O/E/O) converters in forming optical service function chains [19]. Zeng and Fang *et al.* leveraged the advantage of flexible spectrum allocation from EON to further considered the scenario of IDC-EONs, and investigated how to achieve efficient spectrum and IT resource orchestration with both tree/chain-type VNF arrangements [20, 21]. The provisioning of VNF graphs in multi-domain IDC-EONs was then studied in [22]. In [23], we took into account the heterogeneity of application requirements and the fairness among users and presented a mixed-strategy gaming model for realizing incentive-driven VNF-SC provisioning in IDC-EONs. Lately, by incorporating the recent advances in machine learning, Li *et al.* proposed a cognitive VNF-SC provisioning framework for IDC-EONs which can proactively consolidate the deployment of VNFs based on the forecast of the future traffic distribution [24].

Nevertheless, the aforementioned previous works all assumed a single decision-maker for coordinating the resource allocation in DCs and transport networks, which violates the autonomy of each administrative domain. Although there have been a few works reporting distributed designs for virtual network embedding [30] or VNF-SC [31], these works only focused on how to calculate the inter-domain service scheme for each user while the fully distributed architecture employed hinders the joint optimization regarding the services of multiple users. In this context, we previously proposed a multi-broker based architecture that deploys a management plane consisting of multiple market-driven brokers on top of domain managers to provide a more realistic and robust mechanism of operating multi-AS networks [25]. The effectiveness of this architecture in multi-domain software-defined EONs has been proved through extensive inter-domain lightpath provisioning studies [26, 32, 33]. Therefore, it is necessary to exploit how to realize efficient VNF-SC in IDC-EONs with the assist of the multi-broker based architecture.

III. Incentive-Driven VNF-SC Provisioning Framework

In this section, we first show the principle of VNF-SC provisioning in IDC-EONs with an illustrative example. Then, we describe in detail the architecture of a multi-broker based IDC-EON supporting incentive-driven VNF-SC provisioning. Finally, we present the formal problem formulation.

A. VNF-SC Provisioning in IDC-EONs

Fig. 1(a) shows an example for VNF-SC provisioning in IDC-EONs, where user *A* (*e.g.*, a secondary service provider or a research institute) requests for a service function chain consisting of VNF-1 and VNF-2 to steer its traffic from *Node* 1 to *Node* 9. Two service schemes labeled by solid and dashed lines respectively (provided by different brokers), can be used here. The corresponding spectrum allocation on lightpaths is given by Figs. 1(b) and (c). Note that, an optical transponder is required for realizing O/E/O conversion at each intermediate node where the user's traffic should be processed by a VNF. Therefore, the spectrum allocation on each link along path 1-3-6-9 can be done separately due to the spectrum conversion capabilities at *Nodes* 3 and 6, while that on path segment 4-7-9 should follow the spectrum continuity and contiguity constraints [34, 35] for transparent transmission. We can see

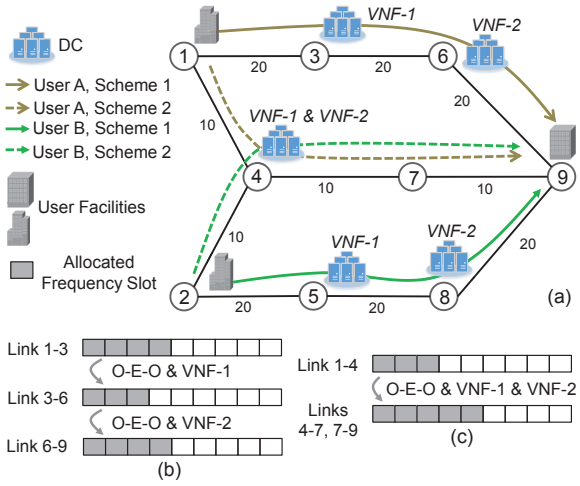


Fig. 1. Example for incentive-driven VNF-SC provisioning in IDC-EONs: (a) case description, (b) and (c) spectrum allocation for paths 1-3-6-9 and 1-4-7-9 respectively.

that different service schemes may result in quite different spectrum and transponder usages. Also, as the VNF processing capacities of DCs differ, the selection of VNF locations can significantly impact the quality-of-service of VNF-SCs, *e.g.*, end-to-end service latency.

B. Network Architecture

Fig. 2(a) depicts the overall architecture of a multi-broker based IDC-EON employing the hierarchical SDN control and management paradigm. The data plane of the IDC-EON consists of a number of DCs inter-connected by the EON (either single-domain or multi-domain EON) for high transmission capacity and flexible bandwidth allocation. In each DC, various kinds of VNFs can be instantiated. Since EONs and DCs are usually managed by different operators, a management plane sitting on top of the domain managers (*i.e.*, EON manager and DC manager) is introduced to coordinate the allocation of cross-stratum resources. More specifically, to overcome the drawbacks of having a single decision-maker in the IDC-EON (*e.g.*, violating the autonomy of administrative domains), we allow multiple brokers (owned by different administrative entities) to reside in the management plane and bridge infrastructure operators and users by providing VNF-SC provisioning services. Basically, brokers can interact with EON and DC managers to collect information regarding the network connectivity and resource utilization, calculate VNF-SC service schemes for users with them, and also communicate with the managers to accomplish the related lightpath and VNF configurations. Note that, each user is able to subscribe to multiple brokers and choose the best service schemes from them to use. Therefore, as shown in Fig. 2(b), the service provisioning interactions among brokers and users actually form a competitive incentive-driven market, where users compete with each other to pursue better services (*e.g.*, lower prices and higher quality-of-service *etc*.) while brokers compete by deploying more advanced provisioning strategies or promising more attractive service commissions for encouraging the use of their services. In all, the proposed IDC-EON architecture enables an incentive-driven service provisioning framework that can not only benefit infrastructure operators from not being dictated by a single network orchestrator but also increase the flexibility of VNF-SC provisioning for users.

Let us recall the example in Fig. 1(a) and consider users *A* and *B* simultaneously. Each user receives two VNF-SC service schemes, which are labeled by solid (from *Broker* 1) and dashed (from *Broker* 2) lines respectively. The numbers on the links indicate the resource costs for using them. We note that although *Broker* 2 offers service schemes with lower costs, the users may not unalterably chose its services because this would induce higher service latencies due to the sharing of the processing capacities of the VNFs. Therefore, each broker and user should carefully decide which service schemes to provide or use in order to achieve higher utilities.

C. Problem Formulation

We consider IDC-EONs consisting of single-domain EONs and multiple DCs and model them as $G(V, E, V_D)$, where V and E represent the node and fiber link sets in G and V_D ($V_D \subseteq V$) is the set of nodes which each is attached by a DC locally [1]. Λ is the set of all the types of VNFs instantiated in V_D while Λ_n is a subset of Λ indicating the types of VNFs available in DC n ($n \in V_D$). A user VNF-SC request can be denoted as $r(s, d, \Gamma, b, T)$, with s and d being the source and destination nodes, Γ containing the demanded types of VNFs in sequence, b being the bandwidth requirement in Gb/s and T representing the service duration. Each request $r_i \in \mathcal{R}$ can receive multiple service schemes $\mathcal{P}_i$ from the set of brokers $\mathcal{B}_i$ it subscribes to. We define the utility that r_i can achieve (similar to the definition of user utility in [36]) by using service scheme $\mathcal{P}_{i,k} \in \mathcal{P}_i$ as,

$$U_{i,k}^{\psi^{-i}} = \frac{\beta_i - c_{i,k}}{\tau_i + D_{i,k}^{\psi^{-i}}}, \tag{1}$$

where β_i is the budget of r_i, $c_{i,k}$ refers to the payment for $\mathcal{P}_{i,k}$, $D_{i,k}^{\psi^{-i}}$ is the end-to-end latency on the service function chain given ψ^{-i} as the set of service schemes used by other requests ($\psi^{-i} = \psi \setminus \psi_i$, where $\psi_i = \mathcal{P}_{i,k}$ if r_i uses $\mathcal{P}_{i,k}$) and τ_i is a parameter conveying the differentiated quality-of-service requirement on service latency. Specifically, by counting the costs of spectrum, transponder and IT resource usages (denoted as SP_u, TR_u and IT_u respectively) of $\mathcal{P}_{i,k}$, we obtain $c_{i,k}$ as,

$$c_{i,k} = (SP_u \cdot p_{SP} + TR_u \cdot p_{TR} + IT_u \cdot p_{IT})(1 + \delta_{i,k}). \tag{2}$$

Here, p_{SP},p_{TR} and p_{IT} represent the unit prices for per period spectrum, transponder and IT resource usages, $\delta_{i,k}$ is the pricing ratio imposed by brokers as the service commission. Meanwhile, we assume $D_{i,k}^{\psi^{-i}}$ consists of the signal propagation time $l_{i,k}$ and the processing time of all the VNFs on the service chain. Let $\varsigma_{n,m}$ be the processing rate of the m-th VNF in DC n and $g_{i,k}^{n,m}$ be a boolean parameter which equals to 1 when $\mathcal{P}_{i,k}$ uses the m-th VNF in DC n. By modeling the

[1]The solution proposed in this work can be easily extended to support a multi-domain EON scenario by allowing brokers calculating service schemes with multi-domain virtual topologies [26].

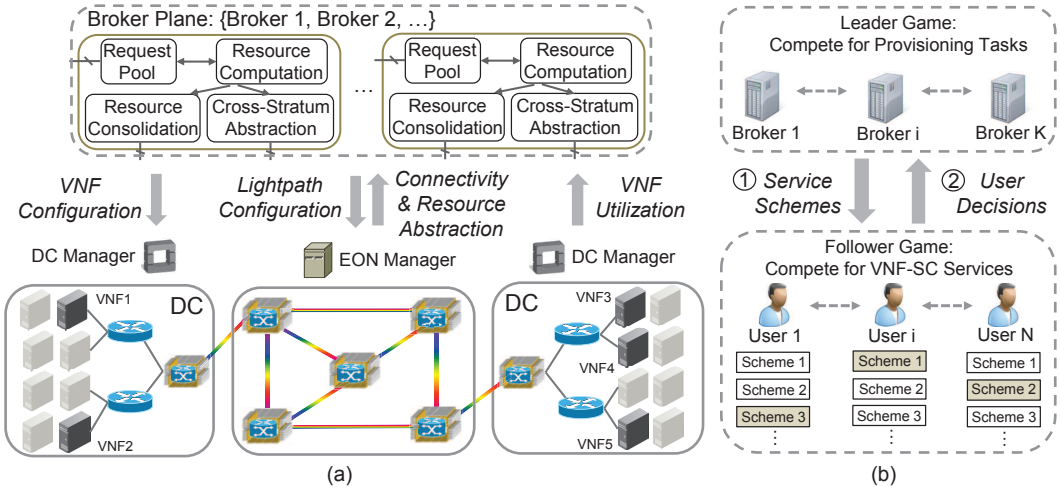


Fig. 2. (a) Proposed broker-based network architecture and (b) interactions among the brokers and users.

processing of each VNF as an M/M/1 model, we can calculate $D_{i,k}^{\psi^{-i}}$ as,

$$D_{i,k}^{\psi^{-i}} = l_{i,k} + \sum_{n\in V_D}\sum_{m\in\Lambda_n} \frac{g_{i,k}^{n,m}}{\varsigma_{n,m} - b_i - \sum_{\mathcal{P}_{t,j}\in\psi^{-i}} g_{t,j}^{n,m} b_t}, \quad (3)$$

$$s.t. \quad \varsigma_{n,m} - g_{i,k}^{n,m}\left(b_i + \sum_{\mathcal{P}_{t,j}\in\psi^{-i}} g_{t,j}^{n,m} b_t\right) > 0, \forall n, m. \quad (4)$$

Note that, Eq. 4 imposes the constraint that the total data rate from users should not exceed the processing rate of each VNF to ensure finite processing time of VNFs. After users having decided the service schemes to use, the utility of each broker ν is determined as the service commissions collected from $\mathcal{R}$, *i.e.*,

$$\mathcal{U}_\nu = \sum_{r_i} \chi_i^\nu c_i^\nu \frac{\delta_i^\nu}{1+\delta_i^\nu}, \quad (5)$$

where χ_i^ν is a boolean variable indicating whether r_i uses service scheme $\mathcal{P}_i^\nu$[2].

In this work, we model the problem of incentive-driven VNF-SC provisioning in IDC-EONs as a noncooperative hierarchical game that consists of a leader game and a follower game. In the leader game, brokers calculate $\mathcal{P}_i$ for each $r_i \in \mathcal{R}$ and compete for the provisioning tasks. After receiving $\mathcal{P}_i$, users play the follower game to decide the service schemes to use for maximizing their utilities defined in Eq. 1. The result of the follower game decides not only the utilities of users, but also the utilities of brokers (as shown in Eq. 5). In the next sections, we will detail the designs for the leader and follower games.

IV. The Follower Game

In this section, we elaborate on the proposed mixed-strategy gaming approach for the follower game and present a time-efficient heuristic algorithm to assist calculating approximate equilibrium solutions. We first investigate the follower game as it is the base of the hierarchical game. Specifically, the optimal strategies of brokers in the leader game depend on the behaviors of both brokers and users, while the follower game only involves users provided the service schemes by brokers.

[2]We reuse the notation $\mathcal{P}$ but with ν being the superscript to denote $\mathcal{P}_i^\nu$ as the service scheme provided by broker ν. The same applies to c_i^ν and δ_i^ν.

TABLE I
Request utilities under different strategy profiles.

	$\mathcal{P}_{2,1}$	$\mathcal{P}_{2,2}$
$\mathcal{P}_{1,1}$	(80, 80)	(80, 140)
$\mathcal{P}_{1,2}$	(140, 80)	(60, 60)

A. Gaming Approach

Nash equilibrium is one of the most important solution concepts that guide players' behaviors in noncooperative games. Conceptually, Nash equilibriums of a game refer to strategy profiles under which no player can increase its utility by unilaterally deviating from them. In other words, every player will follow the strategy indicated by the Nash equilibrium. A strategy profile ψ^* is a pure-strategy Nash equilibrium of the follower game if and only if,

$$U_{i,k}^{(\psi^*)^{-i}} \geq U_{i,j}^{(\psi^*)^{-i}}, \forall \mathcal{P}_{i,k} \in \psi^*, j \neq k. \quad (6)$$

Recall the example in Fig. 1(a), if we assume that $c_{i,k}$ equals to the summation of all the numbers along the corresponding path, $\beta_1 = \beta_2 = 100$, $\tau_1 = \tau_2 = 1/15$, $b_1 = b_2 = 4$, $l_{i,k} = 1/10$ and $\varsigma_{n,m} = 10, \forall i, k, n, m$, we can calculate the utilities of the two users under different strategy profiles as shown in Table I and easily verify that $\{\mathcal{P}_{1,1}, \mathcal{P}_{2,2}\}$ and $\{\mathcal{P}_{1,2}, \mathcal{P}_{2,1}\}$ are two pure-strategy Nash equilibriums of the game. However, the two pure-strategy Nash equilibriums are biased, *i.e.*, always being unfair to one of the users. Hence, it is difficult for the users to decide which equilibrium point to use in real operations. Moreover, not every game is guaranteed to have pure-strategy Nash equilibriums and it is often difficult to prove the existence or calculate them, especially for games with discrete strategy spaces [37]. On the other hand, mixed-strategy gaming, where each player mixes up a number of strategies according to a certain probability distribution other than playing a fixed strategy, provides a useful insight into the study of such games [38]. In this work, we investigate

how to facilitate incentive-driven VNF-SC provisioning with the mixed-strategy game-theoretic approach.

With mixed-strategy gaming, each r_i selects service scheme $\mathcal{P}_{i,k}$ with a probability $x_{i,k} \in [0,1]$, and its objective in the follower game becomes maximizing the expected utility, *i.e.*,

$$\max_{x} \quad U_i(x) = \sum_{\mathcal{P}_{i,k}} x_{i,k} \sum_{\psi^{-i}} U_{i,k}^{\psi^{-i}} \prod_{\mathcal{P}_{t,j} \in \psi^{-i}} x_{t,j}, \tag{7}$$

$$\text{s.t.} \quad \sum_{\mathcal{P}_{i,k}} x_{i,k} = 1. \tag{8}$$

Here, we assume that users make decisions independently and the term $\prod_{\mathcal{P}_{t,j} \in \psi^{-i}} x_{t,j}$ corresponds to the probability that other users' decisions form ψ^{-i}.

The concept of Nash equilibrium is also applicable for mixed-strategy games, leading to mixed-strategy Nash equilibrium which is defined as,

$$U_i\left(x_i^*, (x^*)^{-i}\right) \geq U_i\left(x_i, (x^*)^{-i}\right), \forall r_i, x_i \neq x_i^*, \tag{9}$$

where x^* is the equilibrium solution. We can further deduce from Eq. (9) that all the service schemes with non-zero probabilities (denoted also the support set $\mathcal{S}_i, \forall r_i$) should have the same maximum expected utility, *i.e.*,

$$U_{i,k} = U_{i,j}, \forall x_{i,k}, x_{i,j} > 0, \tag{10}$$

where $U_{i,k} = \sum_{\psi^{-i}} U_{i,k}^{\psi^{-i}} \prod_{\mathcal{P}_{t,j} \in \psi^{-i}} x_{t,j}$. It is easy to verify that if there exists $\mathcal{P}_{i,k}, \mathcal{P}_{i,j} \in \mathcal{S}_i$ with $U_{i,k} > U_{i,j}$, then to achieve a higher utility, r_i will definitely assign a zero probability to $\mathcal{P}_{i,j}$, which contradicts the assumption that $\mathcal{P}_{i,j} \in \mathcal{S}_i$.

Theorem 1. *A game with finite number of players and strategy space has at least one mixed-strategy Nash Equilibrium [39].*

Theorem 1 actually ensures that the game expressed by Eqs. (7)-(8) has at least one mixed-strategy Nash equilibrium. Then, according to [40], we can calculate the Nash equilibrium with the procedures shown in *Algorithm* 1. Basically, we first need to calculate $\mathcal{S}_i$ for each r_i with the iterated-dominance approach depicted in *Lines* 2-9, and then solve the equation set composed by Eqs. (8) and (10) to obtain the probability of using each service scheme. Again, we use the example in Fig. 1(a) with the same parameter assumptions made for Table I to show how mixed-strategy Nash equilibrium can be calculated. Firstly, we calculate $\mathcal{S}_1$ and $\mathcal{S}_2$ both containing two service schemes as neither of the schemes can dominate the other one. For instance, when $x_{2,1} \leq 0.25$, $U_{1,1} \geq U_{1,2}$, otherwise, $U_{1,1} < U_{1,2}$. Then, according to Eq. (10), we obtain the following two equations: (1) $80 = 140 \cdot x_{2,1} + 60 \cdot x_{2,2}$, (2) $80 = 140 \cdot x_{1,1} + 60 \cdot x_{1,2}$. By solving these equations we finally get the equilibrium strategy and expected utilities of the two users as, $x_{1,1} = x_{2,1} = 0.25$, $x_{1,2} = x_{2,2} = 0.75$ and $U_1 = U_2 = 80$.

Note that, since Eq. (10) involves the enumeration of ψ^{-i} ($|\mathcal{R}| + \sum_{r_i} \binom{\mathcal{S}_i}{2}$ equations in total to solve) and a nonlinear operation $\prod x_{t,j}$, the problem becomes intractable when the number of requests exceeds three [41]. Therefore, we will propose a time-efficient heuristic algorithm in the next section to find approximate equilibrium solutions for the game.

Algorithm 1: Procedures of Calculating Mixed-Strategy Nash Equilibrium.

1 set $\mathcal{S} = \mathcal{P}$, $\mathcal{S}' = \emptyset$;
2 **while** $\mathcal{S} \neq \mathcal{S}'$ **do**
3 $\mathcal{S}' = \mathcal{S}$;
4 **for** *each* r_i **do**
5 enumerate all possible ψ^{-i} with $\mathcal{S}$;
6 calculate $\max_{\psi^{-i}} U_{i,k}^{\psi^{-i}}$ and $\min_{\psi^{-i}} U_{i,k}^{\psi^{-i}}$, $\forall \mathcal{P}_{i,k} \in \mathcal{S}_i$;
7 delete $\mathcal{P}_{i,j}$ from $\mathcal{S}_i$ if $\max_{\psi^{-i}} U_{i,j}^{\psi^{-i}} \leq \min_{\psi^{-i}} U_{i,k}^{\psi^{-i}}, \exists \mathcal{P}_{i,k}$;
8 **end**
9 **end**
10 solve the equation set composed by Eqs. (8) and (10) to get x;

Algorithm 2: Procedures of Calculating $\min_{\psi^{-i}} U_{i,k}^{\psi^{-i}}$ and $\max_{\psi^{-i}} U_{i,k}^{\psi^{-i}}$.

1 set $\psi^- = \psi^+ = \emptyset$;
2 **for** *each* $r_t (t \neq i)$ *in the descending order of* b_t **do**
3 set $\psi^- = \arg \min_{\psi = \psi^- \bigcup \mathcal{P}_{t,j}, \forall j} D_{i,k}^{\psi}$;
4 set $\psi^+ = \arg \max_{\psi = \psi^+ \bigcup \mathcal{P}_{t,j}, \forall j} D_{i,k}^{\psi}$;
5 **end**
6 calculate $\max_{\psi^{-i}} U_{i,k}^{\psi^{-i}} = \frac{\beta_i - c_{i,k}}{D_{i,k}^{\psi^-}}$ and $\min_{\psi^{-i}} U_{i,k}^{\psi^{-i}} = \frac{\beta_i - c_{i,k}}{D_{i,k}^{\psi^+}}$;

B. Heuristic Algorithm

We first design a simple greedy method to accelerate the iterated-dominance approach in *Algorithm* 1. Basically, we aim to obtain the approximations of $\max_{\psi^{-i}} U_{i,k}^{\psi^{-i}}$ and $\min_{\psi^{-i}} U_{i,k}^{\psi^{-i}}$ without enumerating ψ^{-i}. This is in turn equivalent to finding ψ^{-i} that leads to the minimum or maximum $D_{i,k}^{\psi^{-i}}$ according to the definition of user utility in Eq. (1). *Algorithm* 2 shows the principle of the designed method. In *Lines* 2-5, we traverse every r_t ($t \neq i$) to iteratively add service schemes that correspond to the minimum or maximum $D_{i,k}^{\psi^{-i}}$ into ψ^- and ψ^+ respectively. Here, we sort r_t in the descending order of b_t as requests with higher data rate have more critical impacts on the service latency of r_i. The complexity of *Algorithm* 2 is $O(|\mathcal{P} - \mathcal{P}_i|)$. In the rest of the paper, we refer to *Algorithm* 1 as the modified one with Lines 5-6 replaced by *Algorithm* 2. Note that, after having obtained $\mathcal{S}$, we still need to enumerate ψ^{-i} with service schemes in $\mathcal{S}$ when calculating the expected utility of each r_i with Eq. (7). Therefore, we approximate $U_{i,k}$ with the expected utilizations of VNFs as,

$$\tilde{U}_{i,k} = \frac{\beta_i - c_{i,k}}{\tau_i + l_{i,k} + \sum_{n \in V_D} \sum_{m \in \Lambda_n} \frac{g_{i,k}^{n,m}}{\varsigma_{n,m} - b_i - \sum_{\mathcal{P}_{t,j} \in \mathcal{S}^{-i}} g_{t,j}^{n,m} b_t x_{t,j}}}. \tag{11}$$

With the above preliminaries, we next discuss how to calculate approximate mixed-strategy Nash equilibrium for the follower game. We first construct an auxiliary graph (AG) to facilitate our algorithm design and Fig. 3(a) shows the

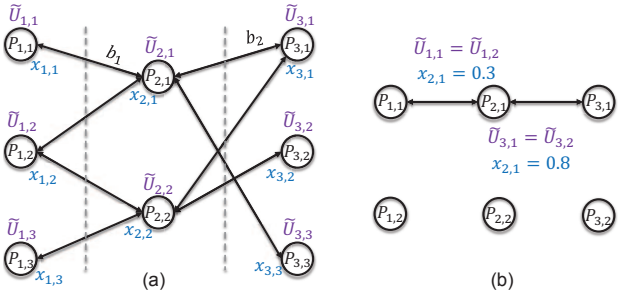


Fig. 3. (a) Principle of the auxiliary graph and (b) an example for the case where mixed-strategy Nash equilibrium does not exist.

principle of the AG. Each node in the AG represents a service scheme $\mathcal{P}_{i,k} \in \mathcal{S}$ and two nodes are connected if the corresponding service schemes share the processing of the same VNFs in certain DCs. Nodes belonging to the same request are not connected as a request finally can only use one service scheme. We assume directed links in the AG and assign each link a weight that is equal to b_i for the reason that service schemes with higher data rate impact more the utilities of their adjacent service schemes.

Algorithm 3 shows the detail of the heuristic design based on the AG. The main idea behind *Algorithm* 3 is to approximate the conditions of mixed-strategy Nash equilibrium defined in Eq. (10) by iteratively adjusting the probability of selecting each $\mathcal{P}_{i,k}$ according to the connectivity in the AG. *Lines* 1-3 are for initialization and we assign equal probabilities to service schemes of each r_i. In each iteration of the optimization, we first estimate the expected utility of each $\mathcal{P}_{i,k} \in \mathcal{S}_i$ with Eq. (11) (*Line* 6), and calculate its deviation from the average value of r_i (*Lines* 7-8). The optimization process terminates when the maximum deviation is less than a preset threshold η_0 (*e.g.*, 0.5%). Otherwise, as shown by *Lines* 12-26, for each $\mathcal{P}_{i,k}$, we increase or decrease the probabilities of its adjacent service schemes in the AG based on whether its expected utility is higher or lower than the average. By doing this, we try to eliminate the utility difference among the service schemes of a request. Here, we set the step size as $\epsilon\alpha^{\left|E_i-\tilde{U}_{i,k}\right|/E_i}$, where ϵ and α are both constant parameters, to achieve adaptive probability adjusting according to the gap with the equilibrium point, *i.e.*, $E_i - \tilde{U}_{i,k} = 0, \forall \mathcal{P}_{i,k} \in \mathcal{S}$. Note that, the iterated dominance approach in *Algorithm* 1 does not necessarily generate a support $\mathcal{S}$ that ensures the existing of mixed-strategy Nash equilibrium, which can be seen through the example in Fig. 3(b). Here, although none of the service schemes is dominated, there is conflict between r_1 and r_3. Specifically, $x_{2,1}$ has to be equal to 0.3 and 0.8 respectively to ensure $\tilde{U}_{1,1} = \tilde{U}_{1,2}$ and $\tilde{U}_{3,1} = \tilde{U}_{3,2}$. Hence, the aforementioned optimization process will be stuck at some point with $0.3 < x_{2,1} < 0.8$, and we need to remove at least one node from the AG to obtain an equilibrium solution. Specifically for this example, there are in total six candidate subsets of $\mathcal{S}$ to be considered. However, it is usually impossible to enumerate and check all the subsets when the number of service schemes is large. Instead of considering all subsets of $\mathcal{S}$, we try to generate feasible and promising subsets by iteratively removing from $\mathcal{S}$ the service scheme that has the largest utility gap compared with the best scheme of the same request (*Line* 29). This is actually motivated by the observation that preferentially removing those service schemes with relatively low utilities facilitates higher total utility of the equilibrium solution. Moreover, according to Schelling's theory on focal point [42], this operation is also likely to be accepted by users in practical scenarios since it caters to a focal point regarding fairness. Finally, in *Lines* 30-31, we recalculate $\mathcal{S}$, unify the probability distribution and proceed to the next optimization process until the optimization goal is reached. The complexity of *Algorithm* 3 is $O\left(Q\left|V\right|\left|\Lambda\right|\left|\mathcal{P}\right|^3\right)$.

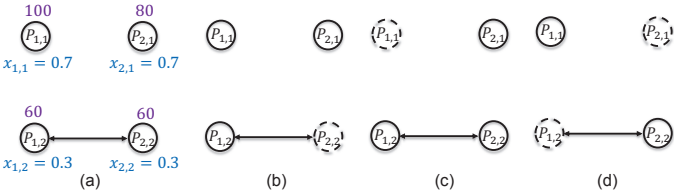


Fig. 4. An example for the broker game: AGs constructed based on different strategy profiles of brokers.

V. The Leader Game

Under the assumption that brokers are aware of the principle of the follower game, we study the leader game in this section. Recall that in the leader game, each broker calculates the most appropriate set of service schemes for users so that its utility (defined by Eq. (5)) is maximized. We first analyze the Nash equilibrium of the game. Fig. 4 gives an example of the leader game. We consider the case where three brokers and two users compete with each other. Fig. 4(a) shows the AG incorporating all the feasible service schemes for the two users, with the number on top of each node representing the utility that a broker can achieve by getting the corresponding service scheme accepted. We also show the probability distribution of user selection (*e.g.*, $x_{1,1} = 0.7$) when all these service schemes are provided to the users. Although providing $(\mathcal{P}_{1,1}, \mathcal{P}_{2,1})$ corresponds to the highest utility expectation according to this AG, it is not the best provisioning strategy for the brokers, since different strategies played by the brokers may result in different AGs (as illustrated in Figs. 4(b)-(d)). For instance, Fig. 4(b) shows the AG constructed for the case when one broker provides $(\mathcal{P}_{1,2}, \mathcal{P}_{2,1})$ while the other two brokers provide $(\mathcal{P}_{1,1}, \mathcal{P}_{2,1})$. Apparently, $\mathcal{P}_{1,1}$ will be dominated by $\mathcal{P}_{1,2}$ as none of them provides $\mathcal{P}_{2,2}$, *i.e.*, providing $(\mathcal{P}_{1,1}, \mathcal{P}_{2,1})$ yields lower utility. In fact, there is no pure-strategy Nash equilibrium in this game. Hence, a mixing of these provisioning strategies becomes a promising yet practical solution for brokers. Recall that $\mathcal{P}^\nu$ denotes a feasible provisioning strategy for broker ν (*e.g.*, $\mathcal{P}^\nu = \{\mathcal{P}_{1,1}, \mathcal{P}_{2,2}\}$) and let $\hat{x}_{\mathcal{P}^\nu}$ be the probability with which broker ν provides $\mathcal{P}^\nu$, the utility function of broker ν in the mixed-strategy game then can be expressed as,

$$\mathcal{U}_\nu = \sum_{\mathcal{P}^\nu} \hat{x}_{\mathcal{P}^\nu} \sum_{\mathbb{P}^{-\nu}} \left(\mathcal{U}_\nu^{(\mathcal{P}^\nu, \mathbb{P}^{-\nu})} \prod_{\mathcal{P}^{\nu'} \in \mathbb{P}^{-\nu}} \hat{x}_{\mathcal{P}^{\nu'}} \right), \qquad (12)$$

where $\mathbb{P}^{-\nu} = \left\{\mathcal{P}^1, ..., \mathcal{P}^{\nu-1}, \mathcal{P}^{\nu+1}, ..., \mathcal{P}^{|\mathcal{B}|}\right\}$. Note that, to calculate the mixed-strategy Nash equilibrium for the leader

Algorithm 3: Procedures of Calculating Approximate Nash Equilibrium.

1 calculate $\mathcal{S}$ with *Lines* 1-9 of *Algorithm* 1;
2 construct an AG based on service schemes in $\mathcal{S}$;
3 initiate $x_{i,k} = 1/\left|\mathcal{S}_i\right|, \forall \mathcal{P}_{i,k} \in \mathcal{S}$;
4 **while** 1 **do**
5 **for** $q = 1 : Q$ **do**
6 calculate $\tilde{U}_{i,k}, \forall \mathcal{P}_{i,k} \in \mathcal{S}$ with Eq. (11);
7 calculate $E_i = \sum_{\mathcal{P}_{i,k} \in \mathcal{S}_i} \tilde{U}_{i,k} / \left|\mathcal{S}_i\right|, \forall r_i$;
8 calculate $\eta_i = \max_{\mathcal{P}_{i,k} \in \mathcal{S}_i} \left|\tilde{U}_{i,k} - E_i\right| / E_i, \forall r_i$;
9 **if** $\eta_i < \eta_0, \forall r_i$ **then**
10 **return;**
11 **end**
12 **for** *each* $\mathcal{P}_{i,k} \in \mathcal{S}$ **do**
13 **if** $\tilde{U}_{i,k} > E_i$ **then**
14 **for** *each adjacent node* $\mathcal{P}_{t,j}$ *of* $\mathcal{P}_{i,k}$ *in AG* **do**
15 set $x_{t,j} = x_{t,j} + \epsilon \alpha^{(\tilde{U}_{i,k} - E_i)/E_i}$;
16 **end**
17 **else if** $\tilde{U}_{i,k} < E_i$ **then**
18 **for** *each adjacent node* $\mathcal{P}_{t,j}$ *of* $\mathcal{P}_{i,k}$ *in AG* **do**
19 **if** $x_{t,j} > \epsilon \alpha^{(E_i - \tilde{U}_{i,k})/E_i}$ **then**
20 set $x_{t,j} = x_{t,j} - \epsilon \alpha^{(E_i - \tilde{U}_{i,k})/E_i}$;
21 **else**
22 set $x_{t,j} = 0$;
23 **end**
24 **end**
25 **end**
26 **end**
27 set $x_{i,k} = \frac{x_{i,k}}{\sum_{\mathcal{P}_{t,j} \in \mathcal{S}_i} x_{t,j}}, \forall \mathcal{P}_{i,k} \in \mathcal{S}$;
28 **end**
29 delete $\arg\max_{\mathcal{P}_{i,k}} \frac{\max_{\mathcal{P}_{i,j} \in \mathcal{S}_i} \tilde{U}_{i,j} - \tilde{U}_{i,k}}{\max_{\mathcal{P}_{i,j} \in \mathcal{S}_i} \tilde{U}_{i,j}}, \forall r_i$ from $\mathcal{S}$;
30 set $\mathcal{S}' = \emptyset$, recalculate $\mathcal{S}$ with *Lines* 2-9 of *Algorithm* 1 and restructure the AG accordingly;
31 set $x_{i,k} = \frac{x_{i,k}}{\sum_{\mathcal{P}_{t,j} \in \mathcal{S}_i} x_{t,j}}, \forall \mathcal{P}_{i,k} \in \mathcal{S}$;
32 **end**

Algorithm 4: Procedures of Calculating Mixed Provisioning Strategies for Broker ν

1 calculate M service schemes for each r_i;
2 run *Algorithm* 3 to get x and $\mathcal{S}$;
3 construct an AG based on $\mathcal{S}$;
4 calculate the weight of each $\mathcal{P}_{i,k}$ in AG as $\omega_{i,k} = \frac{x_{i,k} c_{i,k}}{\max_{\mathcal{P}_{i,j}} c_{i,j}} \frac{\delta_{i,k}}{1+\delta_{i,k}}$;
5 $\mathbb{P} = \emptyset$;
6 **while** $\max \omega > 0$ **do**
7 $\mathcal{P}^\nu = \emptyset$;
8 $\varrho = \mathbf{1}_{1\mathrm{x}|\mathcal{R}|}$;
9 store $\mathcal{P}_{i,k} = \arg\max_{\mathcal{P}_{t,j} \in \mathcal{S}} \omega_{t,j}$ in $\mathcal{P}^\nu$;
10 $\omega_{i,k} = \omega_{i,k} - C_0$;
11 $\varrho_i = 0$;
12 $\mathcal{S}' = \mathcal{S}_i - \mathcal{P}_{i,k}$;
13 **while** $\varrho_t > 0, \exists r_t$ **do**
14 $\omega'_{i,k} = \omega_{i,k} + \sum_{\mathcal{P}_{t,j} \in \mathcal{S}'} \mathrm{AG}_{\mathcal{P}_{i,k}, \mathcal{P}_{t,j}}, \forall \mathcal{P}_{i,k} \in \mathcal{S} \& \varrho_i > 0$;
15 store $\mathcal{P}_{i,k} = \arg\max_{\mathcal{P}_{t,j} \in \mathcal{S} \& \varrho_j > 0} \omega'_{t,j}$ in $\mathcal{P}^\nu$;
16 $\mathcal{S}' = \{\mathcal{S}', \mathcal{S}_i - \mathcal{P}_{i,k}\}$;
17 $\varrho_i = 0$;
18 $\omega_{i,k} = \omega_{i,k} - C_0$ if $\omega_{i,k} > 0$;
19 **end**
20 $\mathbb{P} = \{\mathbb{P}, \mathcal{P}^\nu\}$;
21 **end**
22 $\hat{x}_{\mathcal{P}^\nu} = \frac{\exp\left(\theta \sum_{\mathcal{P}_{i,k} \in \mathcal{P}^\nu} \left(\omega_{i,k} + C_0\right)\right)}{\sum_{\mathcal{P}'^\nu \in \mathbb{P}} \exp\left(\theta \sum_{\mathcal{P}_{i,k} \in \mathcal{P}'^\nu} \left(\omega_{i,k} + C_0\right)\right)}, \forall \mathcal{P}^\nu \in \mathbb{P}$;

game, we need to explore the whole space of strategy profiles, which is intractable for scenarios with larger numbers of brokers and users. For example, for a leader game with $|\mathcal{B}|$ brokers, N users and M feasible service schemes for each user, we need to run *Algorithm* 3 for $O\left(M^{N|\mathcal{B}|}\right)$ times to simply calculate all $\mathcal{U}_\nu^{\left(\mathcal{P}^\nu, \mathbb{P}^{-\nu}\right)}$. In this work, we propose a time-efficient heuristic algorithm to facilitate brokers achieving higher utilities.

Algorithm 4 shows the detail of the procedures of calculating mixed provisioning strategies for broker ν. We first calculate the space of service schemes containing M schemes per request in *Line* 1. Based on the assumption that brokers know the gaming approach of users, *Lines* 2-3 run *Algorithm* 3 to get the probability distribution with which users will select each of these schemes and construct an AG accordingly. Although the obtained probability distribution might be different from the actual one used by users subsequently, it provides a useful insight for analyzing the behaviors of users. Then, we assign each node in the AG a weight that is equal to the expected utility a broker can achieve by providing it in *Line* 4. Instead of enumerating all possible provisioning strategies, the while-loop covering *Lines* 6-21 iteratively calculates a set of most promising provisioning strategies (*i.e.*, $\mathbb{P}$) until all service schemes in $\mathcal{S}$ are visited. Basically, in each loop of calculating a new provisioning strategy $\mathcal{P}^\nu$, the algorithm starts with putting the service scheme with the highest weight in $\mathcal{P}^\nu$ and updates its weight by subtracting a relatively large constant C_0 from it to encourage the selection of unvisited service schemes (*Lines* 9-10). We use another while-loop (*Lines* 13-19) to include service schemes for all the requests in $\mathcal{P}^\nu$. Here, we build a set $\mathcal{S}'$ with all the service schemes of already included requests except for those in $\mathcal{P}^\nu$. *Lines* 14-15 select the service scheme that has the largest total link weight in the AG with the service schemes from $\mathcal{S}'$. The reason for this operation is that by selecting a service scheme that has the maximum sharing of VNF processing with those in $\mathcal{S}'$, we reduce the probability of service schemes in $\mathcal{P}^\nu$ being dominated. This can be explained by the example in Fig. 4, *i.e.*, it is better to provide $\mathcal{P}_{2,2}$ which shares VNF processing with $\mathcal{P}_{1,2}$ to r_2 if the broker has already decided to provide $\mathcal{P}_{1,1}$. Finally, *Line* 22 applies the Boltzmann function and generates

for each provisioning strategy in $\mathbb{P}$ a probability according to its total expected utility. Here, θ is a parameter for adjusting the shape of the probability distribution, *e.g.*, broker ν will take uniformly random decisions when $\theta = 0$. The optimal setup of θ depends on the actual situations of the leader game and the follower game, such as the number of brokers involved, the provisioning policies they apply and the number of users, and therefore is difficult to be formulated. On the other hand, by analyzing results of historical games and successively adapting the setup of θ, brokers can potentially learn the most appropriate provisioning policy. Specifically, let Θ denote the set containing all the discrete and feasible values of θ, broker ν can start with the random selection of $\theta \in \Theta$ and archive the user acceptance ratio (the ratio of the number of users that accept the service schemes from broker ν to the total number of users) regarding each selection. After sufficient amount of repeated games, a ε-greedy approach can be employed, which makes a random selection of θ with a probability ε and selects θ corresponding to the highest acceptance ratio otherwise. The more games broker ν plays, the more accurate approximation of the optimal θ it can achieve. The complexity of *Algorithm* 4 is $O\left(\left(Q\left|V\right|\left|\Lambda\right|+\left|\mathcal{R}\right|\right)\left|\mathcal{P}\right|^3\right)$.

VI. Simulation Results

In this section, we evaluate the performance of the designed game-theoretic approaches through numerical simulations.

A. Simulations for the Follower Game

We first assume that users already receive the set of service schemes from brokers and conduct simulations to investigate the follower game. Table II summarizes the setup of the parameters used in the simulations[3]. We consider two topologies, *i.e.*, the 14-node NSFNET topology and the 11-node COST239 topology, and apply the same setup of the parameters for both of the topologies except that V_D and the link capacity are different. In addition, we assume that each VNF-SC request r_i randomly demands two types of VNFs and β_i is given based on the estimation of the resource cost on the longest path for the request.

We perform simulations to evaluate the accuracy of the utility estimation method in Eq. 11. Basically, we set the number of VNF-SC requests as 100, calculate service schemes for each of the requests and randomly initialize a probability distribution for them. Then, we compare the utilities calculated based on Eq. 11 (U_{est}) with those from Monte Carlo simulations (U_{sim}). Here, in each simulation, we make every request chose a service scheme based on the generated probability distribution, and average the results from 10000 independent simulations to obtain the expected utility of using each service scheme. Table III presents the results of the estimation error from Eq. 11 (defined as the relative difference between U_{est} and U_{sim}) with the NSFNET topology and different VNF processing rates $\varsigma_{n,m}$. We can observe that the proposed method can achieve relatively good estimations of the utilities with the largest difference being only 7.54%.

[3]We tested different setups of the parameters and Table II presents the one with which the algorithms perform the best.

TABLE II
Simulation setup.

$G(V,E)$	14-node NSFNET Topology & 11-node COST239 Topology [35]
V_D (NSFNET Topology)	Nodes {1, 4, 6, 7, 9, 11, 14}
V_D (COST239 Topology)	Nodes {2, 4, 5, 6, 9, 10}
Link Capacity (NSFNET Topology)	358 Frequency Slots
Link Capacity (COST239 Topology)	180 Frequency Slots
$\Lambda_n = \Lambda$	VNFs {1, 2, 3, 4, 5, 6}
$\left\|\mathcal{P}_i\right\|$	10
b_i	[25, 250] Gb/s
τ_i	$[0.01, 0.05]$
$\delta_{i,k}$	0.05
$\left[p_{SP}, p_{TR}, p_{IT}\right]$	[10, 50, 1] Units
$[Q, \eta_0, \epsilon, \alpha]$	$[300, 0.5\%, 0.008, 20]$
Θ	$\{0, 5, 10, 15, 20, 25\}$
ε	0.02

We then investigate the convergence of *Algorithm* 3 and Table III shows the iterations of the optimization process covering *Lines* 5-28 needed for the algorithm to converge. The reason why the convergence time increases with the decreasing of $\varsigma_{n,m}$ is that when the VNF processing rate decreases, service latencies play more significant roles in determining the utilities of users (refer to Eqs. 1 and 3), resulting in service schemes with higher resource costs more unlikely to be dominated and thereby a larger initial support set $\mathcal{S}$. Note that, since connection requests in optical networks usually arrive much slower (*e.g.*, with the time intervals being tens of minutes or even hours) compared with packet networks, the number of requests to be processed in batch generally would be much smaller than 100 in real operations.

TABLE III
Results of the estimation error from Eq. 11 and the convergence of *Algorithm* 3 (NSFNET topology).

$\varsigma_{n,m}$ (Gb/s)	2000	1800	1600	1400	1200
$\frac{\left\|U_{\text{est}}-U_{\text{sim}}\right\|}{U_{\text{sim}}}$ (%)	0.49	1.04	2.56	4.36	7.54
Iterations to Converge	2100	24600	56400	86700	110100

Next, we conduct simulations to compare the performance of the proposed game-theoretic approach for users (VNF-SC-Game) with those of two baseline algorithms, namely, VNF-SC-LC and VNF-SC-Random, each of which selects service schemes with the least resource cost or randomly. Figs. 5-7 show the simulation results with the NSFNET topology and different $\varsigma_{n,m}$. We can observe that VNF-SC-Game facilitates the highest user utility in all the scenarios, while VNF-SC-Random performs the worst. This is because with the designed gaming approach, users can intelligently adjust the probability of using each service scheme to achieve a good balance between resource consumptions and end-to-end latencies. Figs. 5(a)-7(a) also show a clear trend of performance gain from VNF-SC-Game against the baseline algorithms when we increase the number of requests or reduce the processing rate of VNFs. The underlying rationale for this is that inappropriate selections of service schemes can more probably result in high service latencies (due to the overuse of VNFs) or even resource contentions as the IDC-EON gets more saturated. Results on average resource consumption cost and end-to-end service latency per request verify the above

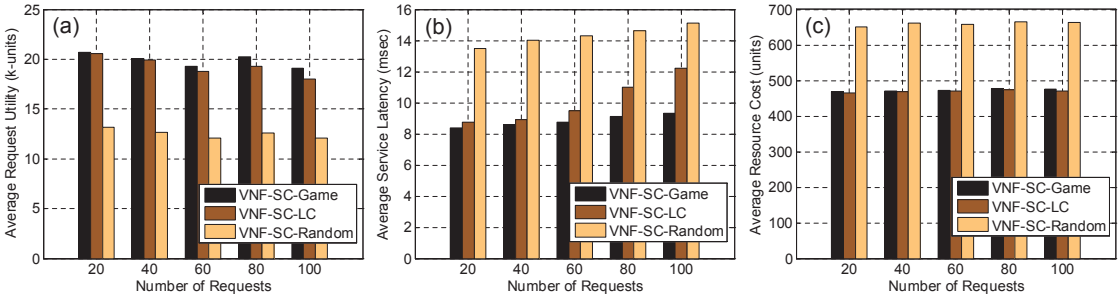


Fig. 5. (a) Request utility, (b) end-to-end service latency and (c) resource consumption cost per request (NSFNET topology, $\varsigma_{n,m} = 1600$ Gb/s).

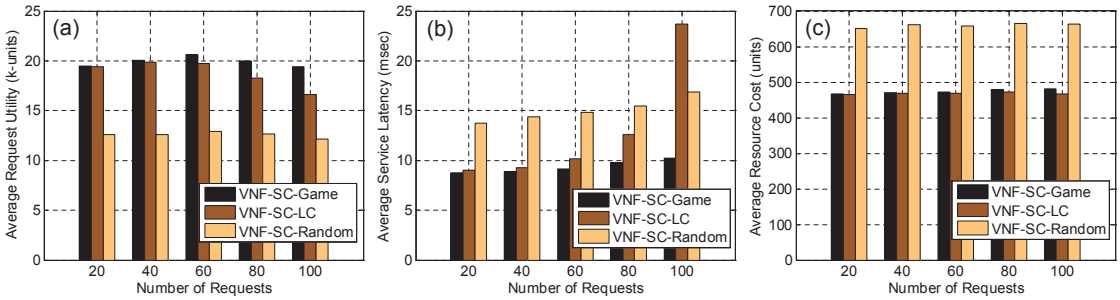


Fig. 6. (a) Request utility, (b) end-to-end service latency and (c) resource consumption cost per request (NSFNET topology, $\varsigma_{n,m} = 1400$ Gb/s).

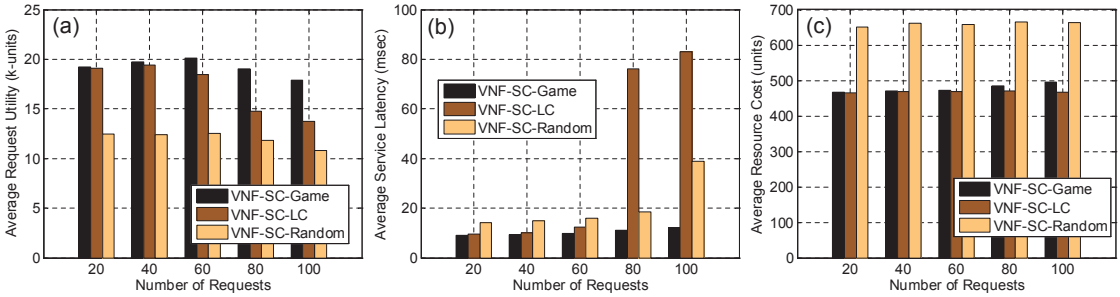


Fig. 7. (a) Request utility, (b) end-to-end service latency and (c) resource consumption cost per request (NSFNET topology, $\varsigma_{n,m} = 1200$ Gb/s).

conjectures. Specifically, we can see that VNF-SC-Game achieves the lowest service latencies while still maintaining comparable resource consumptions with those of VNF-SC-LC. The average service latency from VNF-SC-LC increases rapidly with the number of requests, especially when the processing rate of VNFs is low. Meanwhile, VNF-SC-Random yields the highest resource consumptions and service latencies since it frequently makes users use relatively long transmission paths. Fig. 8 plots the probability distribution functions (PDFs) of request utility from different algorithms when the number of requests and the processing rate of VNFs are 100 and 1200 Gb/s respectively. Here, we record the utility of every individual request during the simulations and approximate the PDFs using the Gaussian kernel density estimation method [26]. We can observe that compared with VNF-SC-LC and VNF-SC-Random, VNF-SC-Game is less likely to lead to service blocks (zero utility). From the angle of user fairness, VNF-SC-Random performs the best as it corresponds to the smallest utility variance (*i.e.*, utilities concentrate closely on the mean value), while the PDF of VNF-SC-LC is the flattest. We also show the results with the COST239 topology in Fig. 9, where similar trends can be observed.

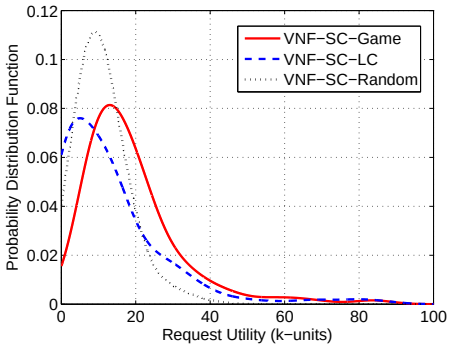


Fig. 8. Distribution of request utility (NSFNET topology, $|\mathcal{R}| = 100$, $\varsigma_{n,m} = 1200$ Gb/s).

B. Simulations for the Leader Game

We perform dynamic service provisioning simulations where VNF-SC requests can come and go on-the-fly for the leader game. All the users are assumed to apply the proposed mixed-strategy gaming approach (*i.e.*, VNF-SC-Game). The processing rate of each VNF is uniformly distributed within $[2800, 3200]$ Gb/s, and the setup for the rest of the parameters is shown in Table II. We denote brokers that provision ac-

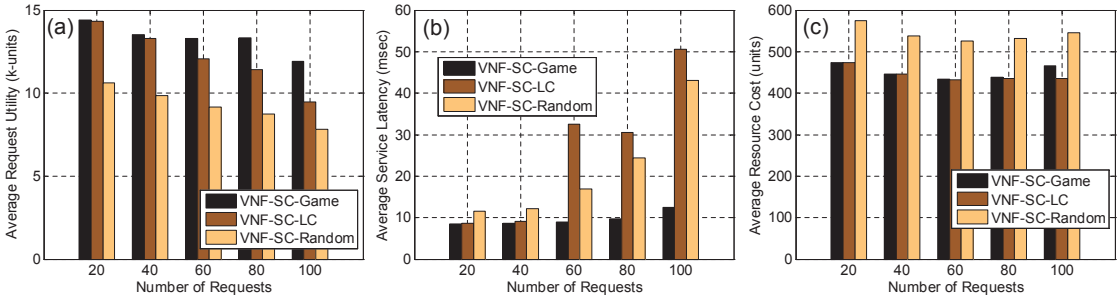


Fig. 9. (a) Request utility, (b) end-to-end service latency and (c) resource consumption cost per request (COST239 topology, $\varsigma_{n,m} = 1200$ Gb/s).

TABLE IV
BROKER UTILITIES PER REQUEST (NSFNET TOPOLOGY).

Traffic Load (Erlangs)	200	250	300	350	400	450	500
BR-GM	14.3	15.1	15.4	15.7	16.0	15.7	15.6
BR-LC	11.4	11.5	11.7	11.6	11.4	11.3	11.1
BR-GM	14.0	14.7	15.2	15.9	15.8	15.6	15.5
BR-LB	11.6	11.9	12.0	11.9	11.7	11.5	11.2
BR-GM	12.9	13.1	13.6	13.6	13.7	13.5	13.5
BR-GM	13.2	13.4	13.6	13.7	13.8	13.6	13.4
BR-GM	10.1	9.8	10.1	10.5	10.5	10.7	10.6
BR-LC	7.9	8.3	8.4	8.3	8.3	8.1	7.9
BR-LB	8.4	8.5	8.7	8.6	8.6	8.4	8.4
BR-GM	8.4	9.0	9.2	9.2	9.1	9.0	9.0
BR-GM	8.6	8.7	8.9	9.2	9.1	9.0	9.0
BR-GM	8.8	8.9	8.9	9.1	9.3	9.0	8.9

TABLE V
BROKER UTILITIES PER REQUEST (COST239 TOPOLOGY).

Traffic Load (Erlangs)	200	250	300	350	400	450	500
BR-GM	12.9	13.3	13.6	13.7	13.8	13.7	13.5
BR-LC	10.2	10.6	10.4	10.3	10.1	9.1	9.6
BR-GM	12.8	13.1	13.3	13.9	13.9	13.7	13.5
BR-LB	10.3	10.7	10.5	10.3	10.1	10.0	9.8
BR-GM	11.6	11.8	11.9	12.0	12.0	11.9	11.6
BR-GM	11.4	12.0	12.0	11.9	12.0	11.8	11.7
BR-GM	8.4	8.7	8.7	9.1	9.2	9.3	9.3
BR-LC	7.4	7.3	7.6	7.4	7.3	7.1	6.8
BR-LB	7.4	7.7	7.7	7.5	7.4	7.2	7.1
BR-GM	7.9	7.8	8.0	8.0	8.0	7.9	7.7
BR-GM	7.9	7.9	7.8	8.0	8.1	7.9	7.7
BR-GM	7.4	7.8	7.9	8.1	8.0	7.9	7.8

cording to the gaming strategy in Algorithm 4 as BR-GM and select BR-LC and BR-LB as the baselines. Specifically, with BR-LC, brokers provide users service schemes from $\mathcal{S}$ with the least resource costs, while BR-LB employs a load balancing strategy that aims to balance the occupation of VNFs across the IDC-EON.

Table IV summarizes the results of broker utility per request with the NSFNET topology when the numbers of brokers are 2 and 3, respectively. We compare the cases where BR-GM plays against BR-LC, BR-LB, BR-GM, or the combinations of them. The results indicate that BR-GM can facilitate much higher broker utility. We can derive similar observations from the results of broker utility with the COST239 topology in Table V, which further verify the superiority of BR-GM. Next, we evaluate the performance of the multi-broker framework by comparing it with the conventional single-broker solutions. Specifically, we employ three brokers with BR-GM for the multi-broker scheme, while the single-broker schemes incorporate BR-LC or BR-LB. Fig. 10(a) plots the results of request blocking probability with the NSFNET topology, showing that the multi-broker scheme can achieve in average $6.5\times$ and $4.8\times$ blocking reductions compared with BR-LC and BR-LB, respectively. This is because the multi-broker scheme offers more service provisioning options for users and in the meantime, the game-theoretic design motivates users to use resources more rationally so that resource bottlenecking is mitigated. Fig. 10(b) shows the average utility achieved by each request. As we can see, the multi-broker scheme facilitates the highest user utility. The overall user utility decreases with the traffic load as the VNFs get more saturated which in turn causes higher end-to-end latencies of VNF-SCs. Fig. 11 shows the results with the COST239 topology. Again, we can see that the multi-broker scheme can remarkably improve

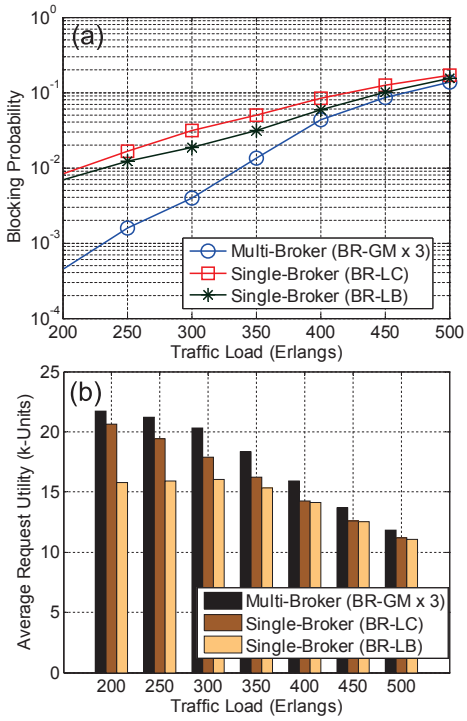


Fig. 10. Comparison between multi-broker and single-broker based schemes: (a) blocking probability and (b) average request utility (NSFNET topology).

the network throughput (*i.e.*, achieving in average $15.7\times$ and $8.6\times$ blocking reductions compared with BR-LC and BR-LB, respectively) and user utility, verifying the robustness of the proposed hierarchical game-theoretic mechanism.

VII. CONCLUSION

In this paper, we proposed for the first time an incentive-driven VNF-SC provisioning framework for multi-broker-

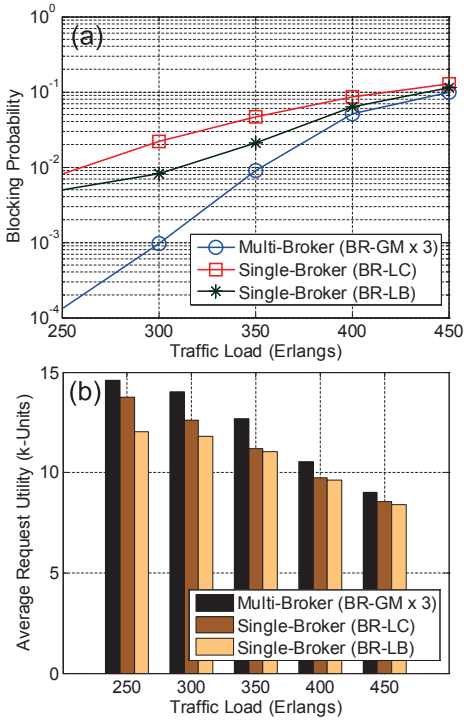


Fig. 11. Comparison between multi-broker and single-broker based schemes: (a) blocking probability and (b) average request utility (COST239 topology).

orchestrated IDC-EONs using game theory. We modeled the problem as a noncooperative hierarchical game where brokers play the leader game, calculate VNF-SC service schemes for users and compete for provisioning tasks from them, while users play the follower game to decide which service schemes from brokers to use so that their utilities are maximized. Mixed-strategy game-theoretic approaches were designed both for brokers and users, with which they could intelligently adjust the use of bandwidth and IT resources according to the network status. Simulation results indicated that the proposed VNF-SC provisioning framework could achieve $> 4.8\times$ reduction on request blocking probability while facilitating higher user and broker utilities compared with the baseline algorithms.

Acknowledgments

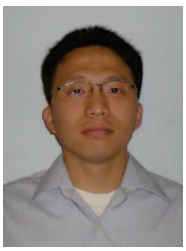
This work was supported in part by DOE DE-SC0016700 and NSF NeTS Award 1619173.

References

[1] "Network functions virtualization (NFV)," *Tech. Rep.*, Oct. 2014. [Online]. Available: https://portal.etsi.org/Portals/0/TBpages/NFV/Docs/NFV_White_Paper3.pdf
[2] R. Mijumbi, J. Serrat, J. Gorricho, N. Bouten, F. Turck, and R. Boutaba, "Network function virtualization: State-of-the-art and research challenges," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 236–262, Firstquarter 2016.
[3] "IETF service function chaining (SFC)," *Tech. Rep.*, Apr. 2014. [Online]. Available: e: https://datatracker.ietf.org/wg/sfc/charter
[4] C. Fung and B. McCormick, "VGuard: A distributed denial of service attack mitigation method using network function virtualization," in *Proc. of CNSM*, Nov 2015, pp. 64–70.
[5] M. Xia, M. Shirazipour, Y. Zhang, H. Green, and A. Takacs, "Optical service chaining for network function virtualization," *IEEE Commun. Mag.*, vol. 53, no. 4, pp. 152–158, 2015.
[6] J. Martins, M. Ahmed, C. Raiciu, V. Olteanu, M. Honda, R. Bifulco, and F. Huici, "ClickOS and the art of network function virtualization," in *Proc. of NSDI*, Apr. 2014, pp. 459–473.
[7] R. Cohen, L. Lewin-Eytan, J. Naor, and D. Raz, "Near optimal placement of virtual network functions," in *Proc. of INFOCOM*, Apr. 2015, pp. 1346–1354.
[8] S. Clayman, E. Maini, A. Galis, A. Manzalini, and N. Mazzocca, "The dynamic placement of virtual network functions," in *Proc. of NOMS*, May 2014, pp. 1–9.
[9] H. Moens and F. Turck, "VNF-P: A model for efficient placement of virtualized network functions," in *Proc. of CNSM*, Nov 2014, pp. 418–423.
[10] B. Addis, D. Belabed, M. Bouet, and S. Secci, "Virtual network functions placement and routing optimization," in *Proc. of CloudNet*, Oct. 2015, pp. 171–177.
[11] T. Lin, Z. Zhou, M. Tornatore, and B. Mukherjee, "Optimal network function virtualization realizing end-to-end requests," in *Proc. of GLOBECOM*, Dec. 2015, pp. 1–6.
[12] M. Luizelli, L. Bays, L. Buriol, M. Barcellos, and L. Gaspary, "Piecing together the NFV provisioning puzzle: Efficient placement and chaining of virtual network functions," in *Proc. of IM*, 2015, pp. 98–106.
[13] R. Mijumbi, J. Serrat, J. Gorricho, N. Bouten, F. De Turck, and S. Davy, "Design and evaluation of algorithms for mapping and scheduling of virtual network functions," in *Proc. of NetSoft*, Apr. 2015, pp. 1–9.
[14] S. Sahhaf, W. Tavernier, D. Colle, and M. Pickavet, "Network service chaining with efficient network function mapping based on service decompositions," in *Proc. of NetSoft*, Apr. 2015, pp. 1–5.
[15] W. Ma, C. Medina, and D. Pan, "Traffic-aware placement of NFV middleboxes," in *Proc. of GLOBECOM*, Dec. 2015, pp. 1–6.
[16] L. Qu, C. Assi, and K. Shaban, "Network function virtualization scheduling with transmission delay optimization," in *Proc. of NOMS*, April 2016, pp. 638–644.
[17] J. Liu, W. Lu, F. Zhou, P. Lu, and Z. Zhu, "On dynamic service function chain deployment and readjustment," *IEEE Trans. Netw. Service Manage.*, vol. 14, no. 3, pp. 543–553, Sept 2017.
[18] S. Zhang, A. Tizghadam, B. Park, H. Bannazadeh, and A. Leon-Garcia, "Joint NFV placement and routing for multicast service on SDN," in *Proc. of NOMS*, April 2016, pp. 333–341.
[19] M. Xia, M. Shirazipour, Y. Zhang, H. Green, and A. Takacs, "Network function placement for NFV chaining in packet/optical datacenters," *J. Lightw. Technol.*, vol. 33, no. 8, pp. 1565–1570, 2015.
[20] M. Zeng, W. Fang, and Z. Zhu, "Orchestrating tree-type VNF forwarding graphs in inter-DC elastic optical networks," *J. Lightw. Technol.*, vol. 34, no. 14, pp. 3330–3341, 2016.
[21] W. Fang, M. Zeng, X. Liu, W. Lu, and Z. Zhu, "Joint spectrum and IT resource allocation for efficient VNF service chaining in inter-datacenter elastic optical networks," *IEEE Commun. Lett.*, vol. 20, no. 8, pp. 1539–1542, 2016.
[22] Y. Wang, P. Lu, W. Lu, and Z. Zhu, "Cost-efficient virtual network function graph (vNFG) provisioning in multidomain elastic optical networks," *J. Lightw. Technol.*, vol. 35, no. 13, pp. 2712–2723, July 2017.
[23] X. Chen, Z. Zhu, J. Guo, S. Kang, R. Proietti, A. Castro, and S. J. B. Yoo, "Leveraging mixed-strategy gaming to realize incentive-driven VNF service chain provisioning in broker-based elastic optical inter-datacenter networks," *J. Opt. Commun. Netw.*, vol. 10, no. 2, pp. 1–9, 2018.
[24] B. Li, W. Lu, S. Liu, and Z. Zhu, "Deep-learning-assisted network orchestration for on-demand and cost-effective vNF service chaining in inter-DC elastic optical networks," *J. Opt. Commun. Netw.*, vol. 10, pp. D29–D41, Oct. 2018.
[25] S. J. B. Yoo, "Multi-domain cognitive optical software defined networks with market-driven brokers," in *Proc. of ECOC*, Sept. 2014, pp. 1–3.
[26] X. Chen, Z. Zhu, L. Sun, J. Yin, S. Zhu, A. Castro, and S. J. B. Yoo, "Incentive-driven bidding strategy for brokers to compete for service provisioning tasks in multi-domain SD-EONs," *J. Lightw. Technol.*, vol. 34, no. 16, pp. 3867–3876, 2016.
[27] B. Han, V. Gopalakrishnan, L. Ji, and S. Lee, "Network function virtualization: Challenges and opportunities for innovations," *IEEE Commun. Mag.*, vol. 53, no. 2, pp. 90–97, 2015.
[28] A. Boubendir, E. Bertin, and N. Simoni, "NaaS architecture through SDN-enabled NFV: Network openness towards web communication service providers," in *Proc. of NOMS*, April 2016, pp. 722–726.
[29] R. Mijumbi, J. Serrat, J. Gorricho, S. Latre, M. Charalambides, and D. Lopez, "Management and orchestration challenges in network functions virtualization," *IEEE Commun. Mag.*, vol. 54, no. 1, pp. 98–105, January 2016.
[30] M. Chowdhury, F. Samuel, and R. Boutaba, "PolyViNE: Policy-based

virtual network embedding across multiple domains," in *Proc. of VISA*, 2010, pp. 49–56.
[31] A. Abujoda and P. Papadimitriou, "DistNSE: Distributed network service embedding across multiple providers," in *Proc. of COMSNETS*, Jan 2016, pp. 1–8.
[32] L. Sun, X. Chen, S. Zhu, Z. Zhu, A. Castro, and S. J. B. Yoo, "Broker-based cooperative game in multi-domain SD-EONs: Nash bargaining for agreement on market-share partition," in *Proc. of ECOC*, Sept. 2016, pp. 1–3.
[33] L. Sun, X. Chen, and Z. Zhu, "Multibroker-based service provisioning in multidomain SD-EONs: Why and how should the brokers cooperate with each other?" *J. Lightw. Technol.*, vol. 35, no. 17, pp. 3722–3733, Sept 2017.
[34] Z. Zhu, W. Lu, L. Zhang, and N. Ansari, "Dynamic service provisioning in elastic optical networks with hybrid single-/multi-path routing," *J. Lightw. Technol.*, vol. 31, no. 1, pp. 15–22, Jan 2013.
[35] X. Chen, S. Zhu, L. Jiang, and Z. Zhu, "On spectrum efficient failure-independent path protection p-cycle design in elastic optical networks," *J. Lightw. Technol.*, vol. 33, no. 17, pp. 3719–3729, 2015.
[36] F. Meshkati, M. Chiang, H. Poor, and S. Schwartz, "A game-theoretic approach to energy-efficient power control in multicarrier CDMA systems," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 6, pp. 1115–1129, June 2006.
[37] S. Sagratella, "Computing all solutions of Nash equilibrium problems with discrete strategy sets," *arXiv preprint arXiv:1512.00653*, 2015.
[38] J. Zhu, B. Zhao, and Z. Zhu, "Leveraging game theory to achieve efficient attack-aware service provisioning in EONs," *J. Lightw. Technol.*, vol. 35, no. 10, pp. 1785–1796, May 2017.
[39] J. Nash, "Non-cooperative games," *Annals of mathematics*, pp. 286–295, 1951.
[40] J. Watson, *Strategy: an introduction to game theory*. WW Norton New York, 2002, vol. 139.
[41] R. Datta, "Using computer algebra to find Nash equilibria," in *Proc. of ISSAC 2003*, 2003, pp. 74–79.
[42] T. Schelling, *The strategy of conflict*. Harvard university press, 1980.

Xiaoliang Chen received his Ph.D. degree from the University of Science and Technology of China in 2016. He is currently a research scholar at UC Davis. His research interests include optical networks, network resilience, software-defined networking and optimization. He is an Associate Editor of Springer Telecommunication Systems Journal and a TPC member of IEEE ICNC 2018, 2019 and IEEE ICC 2018.



Zuqing Zhu received his Ph.D. degree from the University of California, Davis, in 2007. He is currently a full professor at the University of Science and Technology of China. Prior to that, he worked in the Service Provider Technology Group of Cisco Systems, San Jose, California. His research focuses on optical networks, and he received the Best Paper Awards from IEEE ICC 2013, IEEE GLOBECOM 2013, IEEE ICNC 2014, IEEE ICC 2015 and IEEE ONDM 2018.

Roberto Proietti received his M.S. degree in telecommunications engineering from the University of Pisa, Italy, in 2004, and his Ph.D. in fiber optical communications from Scuola Superiore Sant'Anna, Pisa, Italy, in 2009. He is a project scientist at UC Davis. His research interests include high-spectrum-efficiency coherent transmission systems and elastic optical networking, access optical networks and radio over fiber, and optical switching technologies and architectures for supercomputing and data center networks.

S. J. Ben Yoo received his B.S. degree in electrical engineering with distinction, his M.S. degree in electrical engineering, and his Ph.D. degree in electrical engineering with a minor in physics, all from the Stanford University, California, in 1984, 1986, and 1991, respectively. He currently serves as a professor of electrical engineering at UC Davis. His research at UC Davis includes optical switching devices, systems, and networking technologies for future computing and communications.