VNF-Based Service Provision in Software Defined LEO Satellite Networks

Ziye Jia, Min Sheng, *Senior Member, IEEE*, Jiandong Li, *Fellow, IEEE*, Di Zhou, *Member, IEEE*, and Zhu Han, *Fellow, IEEE*

***Abstract*— Low earth orbit (LEO) satellite networks will play important roles in the sixth generation (6G) communication system. Software defined network technique is a novel approach introduced to the LEO satellite networks to improve the resource flexibility and efficiency, forming the software defined LEO satellite networks (SDLSNs). How to efficiently allocate the resources of SDLSN to provide services for the terrestrial users is a key issue. Hence, in this work, we explore the service provision for SDLSN via virtual network functions (VNFs) orchestration on the software defined time-evolving graph. In view of the scarce, intermittent and unstable satellite-to-satellite (S2S) links, the problem is formulated to minimize the S2S resource consumption while satisfying the terrestrial tasks, which is in the form of integer linear programming. Since the problem is intractable by exhaustive search, we design a branch-and-price algorithm based on the coupling of Dantzig-Wolfe decomposition, column generation, and branch-and-bound to efficiently acquire the optimal solution. Further, to obtain a faster solution for practical usage, we further design an approximation algorithm for the subproblem and leverage the beam search to accelerate the pruning for the search tree. Finally, extensive simulations are conducted and the numerical results validate the effectiveness of the proposed schemes.**

***Index Terms*— Low earth orbit (LEO) satellite networks, virtual network function (VNF), satellite resource allocation, branch-and-price.**

I. Introduction

SATELLITE networks will play important roles in the sixth generation (6G)[1] communication system to provide global services, especially for the users in remote areas such as oceans and depopulated zones [2], [3]. Compared with the geosynchronous earth orbit (GEO) and medium earth orbit satellites, low earth orbit (LEO)[2] satellites attract massive scientific and industrial interest due to the lower development cost, lower launch cost, and lower delay [4], [5]. For example, SpaceX Starlink plans to launch above 40,000 LEO satellites for the global network service [6]. However, traditional satellites are specialized for particular tasks, without cooperation and resulting in resource under utilization and high operating cost [7]. Hence, to improve the resource flexibility and efficiency, software defined network (SDN) and network function virtualization (NFV) techniques are introduced into LEO satellite networks, forming the software defined LEO satellite networks (SDLSNs). In particular, SDN helps to manage the whole network in a centralized control pattern by separating the data plane and control plane. NFV enables to decouple virtual network functions (VNFs) from the physical resources to provide services for various users [7], [8].

In SDLSN, how to implement the resource allocation for service provision is a key issue. Note that VNF orchestration based resource allocation is an effective mechanism to cope with the matching between tasks and network resources. However, it is still technically challenging to implement VNF orchestration in SDLSN, due to the following reasons:

- To implement VNF orchestration in SDLSN, various resources of SDLSN should be exactly depicted in advance. However, the relative movement between different satellites results in the dynamic network topology and dynamic resource availability, and thus it is intractable to precisely represent the heterogeneous resources and the inter-relationships among these resources.
- Due to the limited satellite payloads, the onboard satellite resources such as computing units, transceivers, and battery capacity are limited, and such resources should be shared and scheduled with high efficiency.
- For the terrestrial tasks, aside from VNF orchestration, routing is coupled with VNFs. Since a routing path for a task should pass all the required VNFs, VNF orchestration locations in turn put restrictions on the routing path. Such a coupling between VNF orchestration and routing complicates the resource allocation for SDLSN. Moreover, the intermittent satellite-to-satellite (S2S) links

Manuscript received September 7, 2020; revised January 10, 2021; accepted April 3, 2021. Date of publication April 16, 2021; date of current version September 10, 2021. This work was supported in part by the Natural Science Foundation of China under Grant U19B2025, Grant 61725103, and Grant 62001347; in part by the China Postdoctoral Science Foundation under Grant 2019TQ0241 and Grant 2020M673344; in part by the Doctoral Students' Short-Term Study Abroad Scholarship Fund of Xidian University; and in part by NSF under Grant EARS-1839818, Grant CNS1717454, Grant CNS-1731424, and Grant CNS-1702850. This article was presented in part at the IEEE ICC 2020, June 2020. The associate editor coordinating the review of this article and approving it for publication was L. Galluccio. *(Corresponding author: Min Sheng.)*

Ziye Jia, Min Sheng, Jiandong Li, and Di Zhou are with the State Key Laboratory of ISN, Xidian University, Xi'an 710071, China (e-mail: ziyejia@stu.xidian.edu.cn; msheng@mail.xidian.edu.cn; jdli@mail.xidian.edu.cn; zhoudi@xidian.edu.cn).

Zhu Han is with the Department of Electrical and Computer Engineering, University of Houston, Houston, TX 77004 USA, and also with the Department of Computer Science and Engineering, Kyung Hee University, Seoul 446-701, South Korea (e-mail: zhan2@uh.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TWC.2021.3072155.

Digital Object Identifier 10.1109/TWC.2021.3072155

[1]The acronyms used in this paper and the corresponding expansions are summarized in Table I.

[2]Since only LEO satellites are considered in the model of this work, we use "satellite" and "LEO satellite" interchangeably in the remaining of the paper.

TABLE I
ACRONYMS AND THE EXPANSIONS

Acronym	Expansion	Acronym	Expansion
6G	sixth generation	O2S	origin to satellite
LEO	low earth orbit	S2D	satellite to destination
GEO	geosynchronous earth orbit	ILP	integer linear programming
SDN	software defined network	CG	column generation
NFV	network function virtualization	DWD	Dantzig-Wolfe decomposition
SDLSN	software defined LEO satellite network	B&P	branch-and-price
VNF	virtual network function	B&B	branch-and-bound
S2S	satellite-to-satellite	MP	master problem
SDTEG	software defined time evolving graph	PP	pricing problem
SDS	software defined satellite	RMP	restricted master problem
NOCC	network operations control center	LRMP	linear programming relaxation of RMP

make the routing in SDLSN more intractable than in ground networks.

In this paper, we concentrate on dealing with the above challenges. Firstly, we employ the software defined time evolving graph (SDTEG) to depict various resources and tasks in SDLSN. SDTEG can characterize the features of periodical and dynamic SDLSN topology, since the storage arcs are introduced to connect the quasi-static network topology in adjacent time slots. Then, we investigate the service provision and resource allocation model for SDLSN by VNF orchestration on SDTEG. In particular, the problem is formulated to minimize the S2S communication resource consumption while satisfying the terrestrial tasks, constrained by the multiple resource limitations and flow restrictions. Since compared with the pre-positioned and fixed computing and storage resources of SDLSN, satellite communication resources are scarce and unstable due to the intermittent S2S links. The problem is in the form of integer linear programming (ILP) and is NP-hard to obtain the optimal solution, especially in the case of large-scale networks [9]. Fortunately, the constraints of the original formulated problem are characterized by a block angular structure after the complicating constraints are removed, and the problem in such structures can be solved by the column generation (CG) [10] while the complicating constraints can be dealt with via the Dantzig-Wolfe decomposition (DWD) [11]. Moreover, to guarantee the optimal solution, the branch-and-price (B&P) algorithm is designed by coupling DWD and CG with the branch-and-bound (B&B) [12]. Since B&P can obtain the optimal solution more efficiently than the brute-force searching, the solution from B&P can serve as a benchmark for other suboptimal methods.

Specifically, by combining DWD and CG, the original problem is equivalently reformulated as two problems with smaller scale: a master problem (MP) and a couple of pricing problems (PPs). At the very beginning, only a fraction of columns are considered for MP, i.e., restricted master problem (RMP), and the linear programming relaxation of RMP (LRMP) and PPs are solved iteratively until the solution of LRMP has no improvement by adding new columns generated from PPs. Moreover, if the final solution of LRMP is not integer, B&P works to branch at an original variable and a new iteration between the updated LRMP and PPs operates. Since in each iteration, only a small part of columns are considered in RMP, and different PPs can be solved in parallel, the total time complexity is considerably reduced. Moreover, to accelerate the solution for practical large scale networks, an approximation algorithm is designed for RMP, and beam search is employed to obtain an effective bound more efficiently.

To recapitulate, the main contributions of this work are summarized as follows.

- We present the VNF based service provision in SDLSN to implement the resource allocation for LEO satellite networks. Furthermore, by employing SDTEG, we propose the detailed model of VNF orchestration on SDLSN, and as far as the authors' knowledge, this is an innovative work focusing on VNF orchestration in satellite networks.
- Since the original formulated problem is intractable to obtain the optimal solution, we propose the B&P algorithm to efficiently obtain the optimal solution, which can serve as a benchmark for other sub-optimal algorithms.
- For more practical applications, we further propose an approximation algorithm for RMP, and employ the beam search when branching, to efficiently obtain a satisfied solution with guaranteed gap.
- Numerical results and corresponding analyses are provided to evaluate the efficiency of proposed algorithms. The complexity of the proposed algorithms is also analyzed.

The skeleton of the paper is organized as follows. The related works are provided in Section II. The system model and problem formulation are presented in Section III. In Section IV, the problem decomposition and corresponding algorithms are provided. Then simulation results and corresponding analyses are provided in Section V. Finally, conclusions are drawn in Section **??**.

II. RELATED WORK

The applications of SDN and NFV in satellite networks have been investigated in some recent works. For instance, in [13], the authors preliminarily explore the integration of SDN/NFV and CubeSat to provide services for global Internet of Things. Reference [14] investigates the use case of SDN-enabled LEO satellite constellation reconfiguration. Reference [15] proposes an SDN-based satellite-ground network architecture in which resources are managed and scheduled on-demand. Authors in [16] design metaheuristic algorithms for the controller

and gateway placement in the SDN-enabled 5G-satellite networks. Reference [17] proposes an integrated space-terrestrial framework to achieve flexible traffic engineering. Moreover, in practice, efforts are also made to enhance the satellite on-board processing ability. European Space Agency, Harris Corporation, and Thales Alenia Space put efforts in the research of software defined satellites (SDSs) [18], which lay the foundation for satellite online processing and flexible resource management. There already exist a couple of SDSs in the space: 1) Tianzhi 1, developed by Chinese Academy of Sciences and China Academy of Space Technology, is an SDS equipped with a cloud computing platform to test the key technologies such as data processing ability in-orbit [19]. 2) Eutelsat Quantum, developed by Eutelsat and European Space Agency and furnished with a software reconfigurable payload, offers flexible in-orbit reconfiguration on coverage, power, frequency and bandwidth [20]. 3) Lockheed Martin's SmartSat is an SDS which enables flexible mission change in-orbit [21]. Besides, the resource of SDS can be allocated to VNF orchestration. For example, on the cloud computing platform of Tianzhi 1, the image processing can be deemed as the VNF orchestration, which consumes the computing resources. However, it lacks specific resource allocation mechanism for SDS, and we present the VNF-based resource allocation for the software defined LEO satellite. Moreover, the existing SDSs work independently while networking among SDSs is the future trend, so the issue of routing should be taken into account.

In regard to routing in LEO satellite networks, there are respectable researches. For example, [22] proposes a collaborative scheme for LEO satellites to offload data by using inter-satellite links. The authors in [23] design a contact graph routing algorithm for the interplanetary network based on the delay tolerant networking. [24] presents two routing metrics to help with extending the service life of LEO satellite networks. In [25], the authors design the algorithms to solve the energy-efficient routing problem for satellite networks. Note that due to the limited onboard processing resources and multiple tasks, the data processing for one task may not be completed by one satellite, so more satellites will be used for data processing, i.e., multiple VNFs being orchestrated on different satellites. Furthermore, since the corresponding data should be transmitted by S2S links, VNFs orchestration is coupled with routing. As far as the authors' knowledge, the research on coupling VNF orchestration and routing in SDLSN is absent in related works, so we mainly deal with such issue in this work.

As above, the resource allocation for an SDS is deemed as simplex VNF orchestration, and the issue of routing for satellite network has sufficient studies. However, as for the VNF orchestration for SDLSN, there are few corresponding literatures. Hence, we refer to the researches on VNF in the terrestrial networks, in which VNF orchestration with routing is a popular strategy for flexible resource management [26]–[28]. A one-shot mapping of virtual network resources is proposed in [26] to increase the virtual network mapping satisfaction ratio. [27] investigates the VNF dynamic implementation to improve network energy efficiency. [28] jointly optimizes service function chain orchestration and readjustment, considering the trade-off between resource and operation consumption. However, since these works are applied to the static networks with fixed resource supply, the proposed methods cannot be directly adopted by the dynamic satellite networks with limited onboard payloads. Consequently, we contribute to design the VNF orchestration model on SDLSN and corresponding algorithms to cope with such challenges.

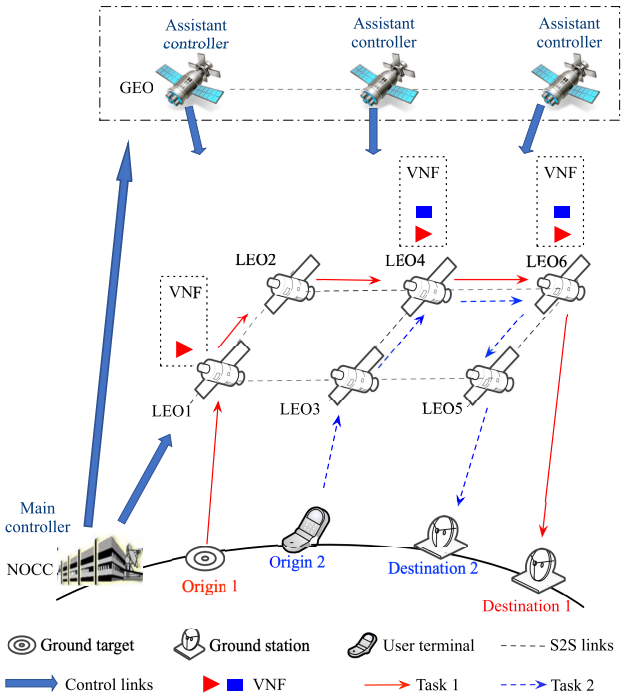


Fig. 1. Scenario of SDLSN with two tasks.

As already noted, both traditional routing algorithms for satellite networks and VNF orchestration strategies for terrestrial networks are not applicable for SDLSN. Our previous work [1] preliminary presents a simplified model and corresponding algorithms for the VNF orchestration in SDLSN. In this work, a more complicated model with specific channel model and energy consumption is presented, an optimal algorithm as well as accelerating algorithms are designed, and more results are provided to reveal the insights.

III. System Model and Problem Formulation

In this section, the SDLSN model, S2S channel model, energy cost model, and the problem formulation are respectively elaborated. Key notations used in this paper are listed in Table II.

A. Network Model

Considering an SDLSN scenario in Fig. 1, the network operations control center (NOCC) on the ground and GEO satellites play as the SDN controller, in which NOCC is the main controller and GEO satellite is the assistant controller. When LEO satellites move out of the sight of NOCC, GEO satellites with global coverage can play as the assistant controller for LEO satellites [29]. LEO satellites are connected by

TABLE II
KEY NOTATIONS

Symbol	Description
$\mathcal{G}(\mathcal{V},\mathcal{E})$	SDTEG composed of nodes $\mathcal{V}$ and links $\mathcal{E}$, $\mathcal{V}=\mathcal{V}_o\cup\mathcal{V}_s\cup\mathcal{V}_d$, $\mathcal{E}=\mathcal{E}_{os}\cup\mathcal{E}_{ss}\cup\mathcal{E}_{sd}\cup\mathcal{E}_s$.
$\mathcal{T}$, T, τ	Time horizon, total number of time slot, time slot length.
$\left(o^t, d^{t'}, \delta_r, \mathcal{F}_r\right)$	A tuple comprised of four elements of task $r\in\mathcal{R}$, including origin node, destination node, data size, and required VNFs $\mathcal{F}_r$, $f_k\in\mathcal{F}_r$.
σ_{f_k}	Amount of computing resource required by VNF f_k.
$\alpha(i^t,j^t)$	Parameter denoting whether link $(i^t,j^t)\in\mathcal{E}_{ss}$ is active.
$c(i^t,j^t)$	Communication capacity of $(i^t,j^t)\in\mathcal{E}_{ss}$.
$\mathcal{C}(i^t,j^t)$	The maximum data size of link $(i^t,j^t)\in\mathcal{E}\backslash\mathcal{E}_s$.
$\mathcal{D}(i^t)$	Computing capacity of node $i^t\in\mathcal{V}_s$.
P_{ss}^{tr}	Transmitting power (in W) for S2S links.
P_{sd}^{tr}	Transmitting power (in W) for S2D links
P_{ss}^{re}	Receiving power (in W) for S2S links.
P_{os}^{re}	Receiving power (in W) for O2S links.
P_o	Power of basic operation (in W) of satellite.
E_s	Energy cost (in J) by a unit of computing resource.
$EB_{i^t}^{max}$	Battery capacity (in KJ).
θ	Maximum discharge degree of battery.
$x_{i^t}^{f_k,r}$	Decision variable in OP. $x_{i^t}^{f_k,r}\in\{0,1\}$ indicates whether VNF $f_k\in\mathcal{F}_r$ is orchestrated on $i^t\in\mathcal{V}_s$.
$y_{i^tj^k}^{r}$	Decision variable in OP. $y_{i^tj^k}^{r}\in\{0,1\}$ indicates whether task r passes $(i^t,j^k)\in\mathcal{E}$.
z_p^r	Decision variable in MP. $z_p^r\in\{0,1\}$ indicates whether task r chooses service path $p\in\mathcal{P}_r$.
$u_{i^t}^{p,f_k}$	Parameter in MP, indicating whether $f_k\in\mathcal{F}_r$ is deployed on $i^t\in\mathcal{V}_s$ for service path $p\in P_r$.
$w_{i^tj^k}^{p}$	Parameter in MP, indicating whether service path $p\in P_r$ passes $(i^t,j^k)\in\mathcal{E}$.
$x_{i^t}^{f_k}$	Decision variable in PP. $x_{i^t}^{f_k}\in\{0,1\}$ denotes whether function $f_k\in\mathcal{F}_r$ of a given task is orchestrated on $i^t\in\mathcal{V}_s$.
$y_{i^tj^k}$	Decision variable in PP. $y_{i^tj^t}\in\{0,1\}$ denotes whether a given task passes $(i^t,j^k)\in\mathcal{E}$.

S2S links and each satellite is equipped with computing and storage resources. There are two types of task in the scenario: task 1 from Origin 1 to Destination 1, and task 2 from Origin 2 to Destination 2. Other than data transmission, the task is also being processed, as VNFs orchestration.[3]

In accordance with Fig. 1, SDTEG in Fig. 2 represents the tasks and resources of SDLSN. In detail, SDTEG is denoted as $\mathcal{G}(\mathcal{V},\mathcal{E})$, where $\mathcal{V}=\mathcal{V}_o\cup\mathcal{V}_s\cup\mathcal{V}_d$ and $\mathcal{E}=\mathcal{E}_{os}\cup\mathcal{E}_{ss}\cup\mathcal{E}_{sd}\cup\mathcal{E}_s$. $\mathcal{V}_o$, $\mathcal{V}_s$, $\mathcal{V}_d$ respectively indicate terrestrial origins, satellites, and terrestrial destinations. $\mathcal{E}_{os}$ and $\mathcal{E}_{sd}$ respectively denote uplinks from terrestrial origins to satellites (O2S) and downlinks from satellites to terrestrial destinations (S2D). Regarding to task $r\in\mathcal{R}$, it is an end-to-end demand and denoted as a tuple $\left\{o^t, d^{t'}, \delta_r, \mathcal{F}_r\right\}$, where o^t, $d^{t'}$, δ_r and $\mathcal{F}_r$ respectively indicate origin, destination, data size and required VNFs. $f_k\in\mathcal{F}_r$ indicates the kth VNF request of $\mathcal{F}_r$, and σ_{f_k} denotes the amount of computing resources consumed by f_k. Due to the periodical movement of satellites, we only explore a time horizon $\mathcal{T}$, which is divided into T time slots, and the length of each time slot is τ. In time slot $t\in T$, the connections among satellites are unchanged and the network topology is quasi-static. Note that $(i^t,j^t)\subset\mathcal{E}_{ss}$ is the S2S link while $(i^t,i^{t+1})\subset\mathcal{E}_s$ is the storage arc of the same satellite in adjacent time slots. Due to the relative movement between different satellites, there may exist no effective S2S links within one time slot, and the data will be stored in the current satellite to the next time slot. However, the connection time length can be predicted according to the periodicity and we use $\alpha(i^t,j^t)$ to denote whether S2S link $(i^t,j^t)\subset\mathcal{E}_{ss}$ is effective, $\alpha(i^t,j^t)=1$ if effective and 0 otherwise.

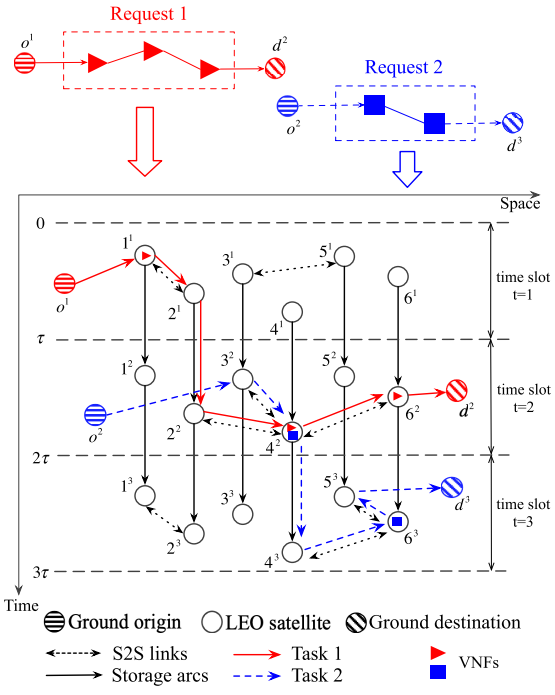


Fig. 2. An illustration of VNF orchestration on SDTEG.

[3]In this work, we concentrate on the resource allocation for NFV orchestration, and for the SDN controller in detail, avid readers are referred to a previous work [7] from our research group.

In addition, due to the low-cost of storage equipments, we do not consider the storage capacity limitation in this paper. Other than routing, the required VNFs $\mathcal{F}_r$ should also be satisfied, as VNFs orchestration in Fig. 2. For instance, task 1 from o^1 to d^2 passes links $(o^1, 1^1)$, $(1^1, 2^1)$, storage arcs $(2^1, 2^2)$, links $(2^2, 4^2)$, $(4^2, 6^2)$, and $(6^2, d^2)$, and $\mathcal{F}_1$ (three triangles) are respectively orchestrated on 1^1, 4^2, and 6^2. Task 2 from o^2 to d^3 passes links $(o^2, 3^2)$, $(3^2, 4^2)$, storage arc $(4^2, 4^3)$, links $(4^3, 6^3)$, $(6^3, 5^3)$, and $(5^3, d^3)$, and $\mathcal{F}_2$ (two squares) are respectively orchestrated on 4^2 and 6^3. It is observed that the computing resource of 4^2 is shared by $f_2 \in \mathcal{F}_1$ and $f_1 \in \mathcal{F}_2$.

B. S2S Channel Model

According to [30], the available data rate of S2S link is expressed as follows,

$$c(i^t, j^t) = \frac{P_{ss}^{tr} G_{tr} G_{re} L_s L_l}{k_B T_s (E_b/N_0)_{req} M}, \tag{1}$$

in which P_{ss}^{tr} is the transmitting power (in W) for S2S links. G_{tr} is the transmitting antenna gain (in dB) and G_{re} is the receiver antenna gain (in dB). L_s is the free space loss (in dB) and L_l (in dB) is the total line loss. k_B is the Boltzmann's constant (in J·K^{-1}). T_s is total system noise temperature (in K). $(E_b/N_0)_{req}$ is the required ratio of received energy-per-bit to noise-density. M is the link margin (in dB) [31]. Additionally, L_s is calculated as

$$L_s = \left(\frac{c}{4\pi \times S \times f} \right)^2. \tag{2}$$

c is light speed (in km/s). S is the maximum slant range (in km). f is the communication center frequency (in Hz). Consequently, the maximum data size of link $(i^t, j^t) \in \mathcal{E}_{ss}$ in time slot t is achieved as

$$\mathcal{C}(i^t, j^t) = \alpha(i^t, j^t) c(i^t, j^t) \tau, \quad \forall (i^t, j^t) \in \mathcal{E}_{ss}, \tag{3}$$

where $\alpha(i^t, j^t)$ defined in Section III-A is a fixed value when the satellite network is given.

C. Energy Cost Model

According to [32], for a satellite $i^t \in \mathcal{V}_s$, energy consumption of the transmitter (S2S, S2D) is expressed as

$$E_{i^t}^{tr} = \sum_{r \in \mathcal{R}} \left(\sum_{j^t \in \mathcal{V}_s} \frac{P_{ss}^{tr} \delta_r y_{i^t j^t}^r}{\mathcal{C}(i^t, j^t)} + \sum_{j^t \in \mathcal{V}_d} \frac{P_{sd}^{tr} \delta_r y_{i^t j^t}^r}{\mathcal{C}(i^t, j^t)} \right) \tau, \tag{4}$$

and energy consumption of the receiver (S2S, O2S) is

$$E_{i^t}^{re} = \sum_{r \in \mathcal{R}} \left(\sum_{j^t \in \mathcal{V}_s} \frac{P_{ss}^{re} \delta_r y_{j^t i^t}^r}{\mathcal{C}(j^t, i^t)} + \sum_{j^t \in \mathcal{V}_o} \frac{P_{os}^{re} \delta_r y_{j^t i^t}^r}{\mathcal{C}(j^t, i^t)} \right) \tau, \tag{5}$$

where P_{sd}^{tr}, P_{ss}^{re}, and P_{os}^{re} are the transmitting power for S2D link, the receiving power for S2S link, and the receiving power for O2S link, respectively. $y_{i^t j^t}^r$ indicates whether r passes $(i^t, j^t) \in \mathcal{E} \backslash \mathcal{E}_s$. δ_r is the data size of task $r \in \mathcal{R}$. Regarding to the energy consumption of VNF, as far as the authors' knowledge, there is no VNF orchestration for satellite networks considering the energy consumption, so we refer to the energy consumption model by extending the terrestrial VNF orchestration from [27],

$$E_{i^t}^F = \sum_{r \in \mathcal{R}} \sum_{f_k \in \mathcal{F}_r} E_c \delta_r \sigma_{f_k} x_{i^t}^{f_k, r}, \quad \forall i^t \in \mathcal{V}_s, \tag{6}$$

where E_c is the energy cost by a unit of computing resource, and $x_{i^t}^{f_k, r}$ indicates whether function $f_k \in \mathcal{F}_r$ is orchestrated on satellite $i^t \in \mathcal{V}_s$. Consequently, the total energy consumption on satellite $i^t \in \mathcal{V}_s$ in time slot t is

$$E_{i^t}^{con} = E_{i^t}^{tr} + E_{i^t}^{re} + E_{i^t}^F + E_{i^t}^o, \tag{7}$$

in which $E_{i^t}^o = P_o \cdot \tau$ is the energy consumption of satellite basic operation.

D. Problem Formulation

1) Flow Constraints: For origin node o^t, each task r from o^t can only choose one satellite $j^t \in \mathcal{V}_s$ for uplink even if there are more satellites within sight of o^t,

$$\sum_{j^t \in \mathcal{V}_s} y_{o^t j^t}^r = 1, \quad \forall r \in \mathcal{R}, \ t \in T. \tag{8}$$

Decision variable $y_{i^t j^t}^r \in \{0, 1\}$ indicates whether task r passes link $(i^t, j^t) \in \mathcal{E} \backslash \mathcal{E}_s$, 1 if passing and 0 otherwise.

Similar to the origin node, for destination node d^t of task r, there is only one satellite $i^t \in \mathcal{V}_s$ can be chosen for downlink,

$$\sum_{i^t \in \mathcal{V}_s} y_{i^t d^t}^r = 1, \quad \forall r \in \mathcal{R}, \ t \in T. \tag{9}$$

As for a middle satellite node i^t, the flow conservation should be satisfied,

$$\begin{cases} \sum\limits_{i^t \in \mathcal{V}_o \cup \mathcal{V}_s} y_{i^t j^t}^r = \sum\limits_{i^t \in \mathcal{V}_s \cup \mathcal{V}_d} y_{j^t i^t}^r + y_{j^t j^{t+1}}^r, \quad \forall r \in \mathcal{R}, j^t \in \mathcal{V}_s, t = 1, \\ \sum\limits_{i^t \in \mathcal{V}_o \cup \mathcal{V}_s} y_{i^t j^t}^r + y_{j^{t-1} j^t}^r = \sum\limits_{i^t \in \mathcal{V}_s \cup \mathcal{V}_d} y_{j^t i^t}^r + y_{j^t j^{t+1}}^r, \\ \qquad \forall r \in \mathcal{R}, j^t \in \mathcal{V}_s, t \in \{2, \ldots, T-1\}, \\ \sum\limits_{i^t \in \mathcal{V}_o \cup \mathcal{V}_s} y_{i^t j^t}^r + y_{j^{t-1} j^t}^r = \sum\limits_{i^t \in \mathcal{V}_s \cup \mathcal{V}_d} y_{j^t i^t}^r, \quad \forall r \in \mathcal{R}, j^t \in \mathcal{V}_s, t = T. \end{cases} \tag{10}$$

In addition, each task r chooses one service path, so the routing does not split,

$$\begin{cases} \sum\limits_{j^t \in \mathcal{V}_s \cup \mathcal{V}_d} y_{i^t j^t}^r + y_{i^t i^{t+1}}^r = 1, \\ \qquad \forall r \in \mathcal{R}, \ i^t \in \mathcal{V}_s, \ t \in \{1, \ldots, T-1\}, \\ \sum\limits_{j^t \in \mathcal{V}_s \cup \mathcal{V}_d} y_{i^t j^t}^r = 1, \quad \forall r \in \mathcal{R}, \ i^t \in \mathcal{V}_s, \ t = T. \end{cases} \tag{11}$$

2) Resource Constraints: Due to the limited S2S communication resource $\mathcal{C}(i^t, j^t)$, the total amount of task passing $(i^t, j^t) \in \mathcal{E} \backslash \mathcal{E}_s$ cannot violate $\mathcal{C}(i^t, j^t)$,

$$\sum_{r \in \mathcal{R}} y_{i^t j^t}^r \delta_r \le \mathcal{C}(i^t, j^t), \quad \forall (i^t, j^t) \in \mathcal{E} \backslash \mathcal{E}_s, \ t \in T. \tag{12}$$

Since the computing resource capacity $\mathcal{D}(i^t)$ of each satellite $i^t \in \mathcal{V}_s$ is limited, the total resource consumed by VNF orchestration on satellite i^t cannot exceed $\mathcal{D}(i^t)$, i.e.,

$$\sum_{r\in\mathcal{R}}\sum_{f_k\in\mathcal{F}_r} x_{i^t}^{r,f_k}\delta_r\sigma_{f_k} \le \mathcal{D}(i^t), \quad \forall i^t \in \mathcal{V}_s,\ t\in T, \tag{13}$$

where decision variable $x_{i^t}^{f_k,r} \in \{0,1\}$ denotes whether VNF $f_k \in \mathcal{F}_r$ is orchestrated on node i^t, 1 if orchestrated and 0 otherwise. It is noted that the amount of computing resource consumption is also affected by the data size δ_r of task r.

Assuming that satellite energy is replenished as $EB_{i^t}^{max}$ at the beginning of each time slot, and the total energy consumption cannot violate the energy capacity,

$$E_{i^t}^{con} \le \theta \cdot EB_{i^t}^{max}, \quad \forall i^t \in \mathcal{V}_s,\ t\in T, \tag{14}$$

where θ is the maximum discharge depth of the battery [24].

3) Other Constraints: Since if the service path of task r passes i^t, VNF $f_k \in \mathcal{F}_r$ is likely to be orchestrated on i^t, and in turn VNF orchestration position has an effect on routing selection. Thus, the coupling constraints between $x_{i^t}^{r,f_k}$ and $y_{i^t j^k}^r$ should be satisfied,

$$\begin{cases} x_{i^t}^{r,f_k} \le \sum\limits_{j^t\in\mathcal{V}_s\cup\mathcal{V}_d} y_{i^t j^t}^r + y_{i^t i^{t+1}}^r, \\ \quad \forall r\in\mathcal{R},\quad f_k\in\mathcal{F}_r,\ i^t\in\mathcal{V}_s,\ t\in\{1,\dots,T-1\}, \\ x_{i^t}^{r,f_k} \le \sum\limits_{j^t\in\mathcal{V}_s\cup\mathcal{V}_d} y_{i^t j^t}^r, \\ \quad \forall r\in\mathcal{R},\quad i^t\in\mathcal{V}_s,\ f_k\in\mathcal{F}_r, t=T. \end{cases} \tag{15}$$

In addition, each function $f_k \in \mathcal{F}_r$ can only be deployed on one satellite node, i.e.,

$$\sum_{i^t\in\mathcal{V}_s} x_{i^t}^{f_k,r} = 1, \quad \forall r\in\mathcal{R},\ f_k\in\mathcal{F}_r,\ t\in T. \tag{16}$$

4) Formulation: Compared with the pre-positioned and fixed computing, storage, energy resources of satellites, S2S communication resources are scarce and unstable due to the intermittent S2S links. Hence, in the original problem (OP), we consider to minimize the total S2S resource consumption while satisfying the given terrestrial tasks, with multiple resource and flow constraints, i.e.,

$$\begin{aligned} \mathbf{OP}:\ \min_{\mathbf{x},\mathbf{y}} & \sum_{t\in T}\sum_{r\in\mathcal{R}(i^t,j^t)\in\mathcal{E}_{ss}}\sum \delta_r y_{i^t j^t}^r \\ \text{s.t. } & (8)-(16), \\ & x_{i^t}^{r,f_k}\in\{0,1\}, \quad \forall r\in\mathcal{R},\ i^t\in\mathcal{V}_s,\ f_k\in\mathcal{F}_r,\ t\in T, \end{aligned} \tag{17}$$

$$y_{i^t j^k}^r \in \{0,1\}, \quad \forall r\in\mathcal{R}, (i^t,\ j^k)\in\mathcal{E},\ t\in T, \tag{18}$$

where $\mathbf{x} = \left\{x_{i^t}^{r,f_k}, \forall r\in\mathcal{R}, i^t\in\mathcal{V}_s, f_k\in\mathcal{F}_r, t\in T\right\}$ and $\mathbf{y} = \left\{y_{i^t j^k}^r, \forall r\in\mathcal{R}, (i^t,j^k)\in\mathcal{E}, t\in T\right\}$.

It is observed that OP is in the form of ILP, and an exhaustive search for the optimal solution is intractable, because the computational complexity of exhaustive search is mainly related to the number of integer variables, the number of total constraints and the scale of the network [33]. In particular, the total number of integer variables is

$$\begin{aligned} N &= |T|\cdot|\mathcal{R}|\cdot|\mathcal{F}|\cdot|\mathcal{V}_s| + |T|\cdot|\mathcal{R}|\cdot|\mathcal{E}| \\ &= (|\mathcal{E}_{os}| + |\mathcal{E}_{ss}| + |\mathcal{E}_{sd}| + |\mathcal{E}_s|) \end{aligned} \tag{19}$$

and the total number of constraints is

$$\begin{aligned} M =& 2\cdot|T|\cdot|\mathcal{R}| + 2\cdot|\mathcal{R}|\cdot|\mathcal{V}_s| + 2\cdot(|T|-1)\cdot|\mathcal{R}|\cdot|\mathcal{V}_s| \\ &+ |T|\cdot|\mathcal{R}|\cdot|\mathcal{V}_s| + |T|\cdot(|\mathcal{E}_{os}| + |\mathcal{E}_{ss}| + |\mathcal{E}_{sd}|) \\ &+ 2\cdot|T|\cdot|\mathcal{V}_s| + |T|\cdot|\mathcal{R}|\cdot|\mathcal{V}_s|\cdot|\mathcal{F}| + |T|\cdot|\mathcal{R}|\cdot|\mathcal{F}| \\ =& |T|\cdot|\mathcal{R}|\cdot(2+3\cdot|\mathcal{V}_s|+|\mathcal{V}_s|\cdot|\mathcal{F}|+|\mathcal{F}|)+|T|\cdot|\mathcal{V}_s|\cdot(4+|\mathcal{V}_s|). \end{aligned} \tag{20}$$

According to [33], the exhaustive search for OP has a time complexity of $\mathcal{O}\left(N\cdot M\cdot 2^N\right)$, which is exponential with the total number of variable N and suffers from the curse of dimensionality. Even for a medium scale OP, the time complexity is unacceptable.

IV. Problem Decomposition and Algorithm Design

We dedicate to deal with OP by employing the Dantzig-Wolfe decomposition and column generation in Section IV-A and IV-B, and the branch-and-price algorithm is designed to obtain the optimal solution in Section IV-C. To further accelerate the solution in practical use, an approximation algorithm is designed in Section IV-D.

A. Dantzig-Wolfe Decomposition

Dantzig-Wolfe decomposition is an efficient method to process the ILP problem with a block angular structure [34]. It is noted that if the complicating constraints (12)-(14) are removed from OP, the residual constraints (8)-(11) and (15)-(18) are in the block angular structure and the reduced problem can be decomposed into $|\mathcal{R}|$ similar subproblems, which can be solved in parallel. Moreover, the polytope $\mathcal{P}$ defined by constraints (8)-(11) and (15)-(18) is a convex hull, which is composed of smaller convex hulls $\mathcal{P}_r$, i.e. $\mathcal{P} = \{\mathcal{P}_r, r\in\mathcal{R}\}$, following the convex hull property [34]. According to Minkowski's Theory [10], a point in the convex hull is equivalent to the linear combination of the extreme points, so we have

$$\left\{ x_{i^t}^{r,f_k} = \sum_{p\in\mathcal{P}_r} z_p^r u_{i^t}^{p,f_k} \mid \sum_{p\in\mathcal{P}_r} z_p^r = 1, z_p^r\in\{0,1\} \right\}, \tag{21}$$

and

$$\left\{ y_{i^t j^k}^r = \sum_{p\in\mathcal{P}_r} z_p^r w_{i^t j^k}^p \mid \sum_{p\in\mathcal{P}_r} z_p^r = 1, z_p^r\in\{0,1\} \right\}, \tag{22}$$

in which $u_{i^t}^{p,f_k}\in\{0,1\}$ and $w_{i^t j^k}^p\in\{0,1\}$ denote the extreme points of the convex hull, while z_p^r indicates the coefficients of these extreme points [35].

To clarify (21) and (22), the corresponding physical meanings are elaborated as follows. $p\in\mathcal{P}_r$ indicates a service path for task r. Apart from a physical path from o^t to $d^{t'}$ in SDTEG, p includes a couple of satellites for VNF $\mathcal{F}_r$ orchestration as

well. $\mathcal{P}_r$ is a set of possible service path for r. $z_p^r \in \{0,1\}$ denotes whether $p \in \mathcal{P}_r$ is selected, and $\sum_{p\in\mathcal{P}_r} z_p^r = 1$ indicates r can only select one service path from $\mathcal{P}_r$. $u_{i^t}^{p,f_k} \in \{0,1\}$ denotes whether $f_k \in \mathcal{F}_r$ is deployed on $i^t \in \mathcal{V}_s$ in service path p, while $w_{i^t j^k}^p \in \{0,1\}$ denotes whether service path p passes link $(i^t, j^k) \in \mathcal{E}$, and such two parameters jointly construct the service path p.

By substituting (21) and (22) into OP, we obtain the master problem (MP) as follows,

$$\mathbf{MP}:$$

$$\min_{\mathbf{z}} \sum_{t\in T}\sum_{r\in\mathcal{R}}\sum_{p\in\mathcal{P}_r}\sum_{(i^t,j^t)\in\mathcal{E}} \delta_r w_{i^t j^t}^p z_p^r$$

$$\text{s.t.} \sum_{p\in\mathcal{P}_r} z_p^r = 1, \quad \forall r \in R, \tag{23}$$

$$\sum_{r\in\mathcal{R}}\sum_{p\in\mathcal{P}_r} z_p^r w_{i^t j^t}^p \delta_r \le \mathcal{C}(i^t, j^t), \quad \forall (i^t, j^t) \in \mathcal{E}\backslash\mathcal{E}_s,\ t \in T, \tag{24}$$

$$\sum_{r\in\mathcal{R}}\sum_{p\in\mathcal{P}_r}\sum_{f_k\in\mathcal{F}_r} z_p^r \sigma_f \delta_r u_{i^t}^{p,f_k} \le \mathcal{D}(i^t), \quad \forall i^t \in \mathcal{V}_s,\ t \in T, \tag{25}$$

$$\sum_{r\in\mathcal{R}}\sum_{p\in\mathcal{P}_r} E_{i^t}^{con,r} z_p^r + E_{i^t}^o \le \theta \cdot EB_{i^t}^{max}, \quad \forall i^t \in \mathcal{V}_s,\ t \in T, \tag{26}$$

$$z_p^r \in \{0,1\}, \quad \forall p \in \mathcal{P}_r,\ r \in \mathcal{R}, \tag{27}$$

where $\mathbf{z} = \left\{z_p^r, \forall r \in R, p \in P_r\right\}$ and

$$E_{i^t}^{con,r} = \left(\sum_{j^t\in\mathcal{V}_s} \frac{P_{ss}^{tr}\delta_r w_{i^t j^t}^p}{\mathcal{C}(i^t,j^t)} + \sum_{j^t\in\mathcal{V}_d} \frac{P_{sd}^{tr}\delta_r w_{i^t j^t}^p}{\mathcal{C}(i^t,j^t)} + \sum_{j^t\in\mathcal{V}_s} \frac{P_{ss}^{re}\delta_r w_{j^t i^t}^p}{\mathcal{C}(j^t,i^t)} + \sum_{j^t\in\mathcal{V}_o} \frac{P_{os}^{tr}\delta_r w_{j^t i^t}^p}{\mathcal{C}(j^t,i^t)}\right)\tau + \sum_{f_k\in\mathcal{F}_r} E_c \delta_r \sigma_{f_k} u_{i^t}^{p,f_k}. \tag{28}$$

Constraints (24)-(26) indicate that the communication resource capacity of $(i^t, j^t) \in \mathcal{E}_{ss}$, computing resource capacity of $i^t \in \mathcal{V}_s$, and energy capacity of $i^t \in \mathcal{V}_s$ cannot be violated, respectively. Note that MP is an equivalent transformation from OP, so the solution of MP is also the solution for OP. Moreover, the linear programming relaxation of MP has a tighter bound than the linear programming relaxation of OP [12], which promotes the solution quality.

B. Column Generation

Intuitively, to solve MP, all service paths $\mathcal{P} = \{\mathcal{P}_r, \forall r \in R\}$ should be obtained in advance. However, it is intractable due to that the number of service paths $|\mathcal{P}|$ grows exponentially with the increasing number of tasks and scale of SDLSN. To cope with this challenge, CG is employed to gradually generate the service paths $\mathcal{P}$. CG is specialized to handle large scale linear and integer programming problems since only a small subset of variables participate in each iteration [36]. At the very beginning, only a small set of service paths $\mathcal{P}' \subseteq \mathcal{P}$ are considered for MP, at least one p for each $r \in \mathcal{R}$, i.e. $\mathcal{P}' = \{\mathcal{P}_r', \forall r \in \mathcal{R}\}$. Thus, the restricted master problem (RMP) is defined as,

$$\mathbf{RMP}:$$

$$\min_{\mathbf{z}} \sum_{t\in T}\sum_{r\in\mathcal{R}}\sum_{p\in\mathcal{P}_r'}\sum_{(i^t,j^t)\in\mathcal{E}} \delta_r w_{i^t j^t}^p z_p^r$$

$$\text{s.t.} \sum_{p\in\mathcal{P}_r'} z_p^r = 1, \quad \forall r \in \mathcal{R}, \tag{29}$$

$$\sum_{r\in\mathcal{R}}\sum_{p\in\mathcal{P}_r'} z_p^r w_{i^t j^t}^p \delta_r \le \mathcal{C}(i^t, j^t), \quad \forall (i^t, j^t) \in \mathcal{E}\backslash\mathcal{E}_s,\ t \in T, \tag{30}$$

$$\sum_{r\in\mathcal{R}}\sum_{p\in\mathcal{P}_r'}\sum_{f_k\in F_r} z_p^r \sigma_f \delta_r u_{i^t}^{p,f_k} \le \mathcal{D}(i^t), \quad \forall i^t \in \mathcal{V}_s,\ t \in T, \tag{31}$$

$$\sum_{r\in\mathcal{R}}\sum_{p\in\mathcal{P}_r'} E_{i^t}^{con,r} z_p^r + E_{i^t}^o \le \theta \cdot EB_{i^t}^{max}, \quad \forall i^t \in \mathcal{V}_s,\ t \in T, \tag{32}$$

$$z_p^r \in \{0,1\}, \quad \forall p \in \mathcal{P}_r',\ r \in \mathcal{R}. \tag{33}$$

RMP is initialized by a set of feasible solution $\mathcal{P}_r'$, and each r has a feasible service path in $\mathcal{P}_r'$. Then, the linear programming relaxation of RMP (LRMP) is solved to get the dual variables $\lambda_r^{(29)}$, $\lambda_{i^t j^t}^{(30)}$, $\lambda_{i^t}^{(31)}$ and $\lambda_{i^t}^{(32)}$, respectively corresponding to constraints (29), (30), (31), and (32), $\boldsymbol{\lambda} = \left\{\lambda_r^{(29)}, \lambda_{i^t j^t}^{(30)}, \lambda_{i^t}^{(31)}, \lambda_{i^t}^{(32)}\right\}$,

$$\mathbf{LRMP}: \min_{\mathbf{z}} \sum_{t\in T}\sum_{r\in\mathcal{R}}\sum_{p\in\mathcal{P}_r'}\sum_{(i^t,j^t)\in\mathcal{E}} \delta_r w_{i^t j^t}^p z_p^r$$
$$\text{s.t. } (29)-(32),$$
$$z_p^r \in [0,1], \quad \forall p \in \mathcal{P}_r', r \in \mathcal{R}.$$

The dual variables $\boldsymbol{\lambda}$ are provided to the pricing problem (PP), and new service paths (columns) are generated by PP. In detail,

$$\mathbf{PP}:$$

$$\min_{\mathbf{x}',\mathbf{y}'} \sum_{t\in T}\sum_{(i^t,j^t)\in\mathcal{E}} \left(1 + \lambda_{i^t j^t}^{(30)}\right)\delta_r y_{i^t j^t} + \sum_{t\in T}\sum_{i^t\in\mathcal{V}_s}\left(\sum_{f_k\in\mathcal{F}_r} \lambda_{i^t}^{(31)}\delta_r\sigma_{f_k} x_{i^t}^{f_k} + \lambda_{i^t}^{(32)} E_{i^t}^{con,r}\right) - \lambda_r^{(29)}$$

$$\text{s.t.} \sum_{j^t\in\mathcal{V}_s} y_{o^t j^t} \le 1, \quad \forall t \in T, \tag{34}$$

$$\sum_{i^t\in\mathcal{V}_s} y_{i^t d^t} \le 1, \quad \forall t \in T, \tag{35}$$

$$\sum_{i^t\in\mathcal{V}_o\cup\mathcal{V}_s} y_{i^t j^t} = \sum_{i^t\in\mathcal{V}_s\cup\mathcal{V}_d} y_{j^t i^t} + y_{j^t j^{t+1}}, \quad \forall j^t \in \mathcal{V}_s,\ t = 1, \tag{36}$$

$$\sum_{i^t\in\mathcal{V}_o\cup\mathcal{V}_s} y_{i^t j^t} + y_{j^{t-1} j^t} = \sum_{i^t\in\mathcal{V}_s\cup\mathcal{V}_d} y_{j^t i^t} + y_{j^t j^{t+1}}, \quad \forall j^t \in \mathcal{V}_s,\ t \in \{2,\ldots,T-1\}, \tag{37}$$

$$\sum_{i^t\in\mathcal{V}_o\cup\mathcal{V}_s} y_{i^t j^t} + y_{j^{t-1} j^t} = \sum_{i^t\in\mathcal{V}_s\cup\mathcal{V}_d} y_{j^t i^t}, \quad \forall j^t \in \mathcal{V}_s,\ t = T, \tag{38}$$

$$\sum_{j^t \in \mathcal{V}_s \cup \mathcal{V}_d} y_{i^t j^t} + y_{i^t i^{t+1}} = 1, \quad \forall i^t \in \mathcal{V}_s,\ t \in \{1, \ldots, T-1\}, \tag{39}$$

$$\sum_{j^t \in \mathcal{V}_s \cup \mathcal{V}_d} y_{i^t j^t} = 1, \quad \forall i^t \in \mathcal{V}_s,\ t = T, \tag{40}$$

$$y_{i^t j^t} \delta_r \le \mathcal{C}(i^t, j^t), \quad \forall (i^t, j^t) \in \mathcal{E} \backslash \mathcal{E}_s,\ t \in T, \tag{41}$$

$$\sum_{f_k \in \mathcal{F}_r} \sigma_f \delta_r x_{i^t}^{f_k} \le \mathcal{D}(i^t), \quad \forall i^t \in \mathcal{V}_s,\ t \in T, \tag{42}$$

$$E_{i^t}^{con,r} + E_{i^t}^{o} \le \theta \cdot EB_{i^t}^{max}, \quad \forall i^t \in \mathcal{V}_s,\ t \in T, \tag{43}$$

$$x_{i^t}^{f_k} \le \sum_{j^t \in \mathcal{V}_s \cup \mathcal{V}_d} y_{i^t j^t} + y_{i^t i^{t+1}}, \quad \forall f_k \in \mathcal{F}_r, i^t \in \mathcal{V}_s, t \in \{1, \ldots, T-1\}, \tag{44}$$

$$x_{i^t}^{f_k} \le \sum_{j^t \in \mathcal{V}_s \cup \mathcal{V}_d} y_{i^t j^t}, \quad \forall i^t \in \mathcal{V}_s,\ f_k \in \mathcal{F}_r,\ t = T, \tag{45}$$

$$\sum_{i^t \in \mathcal{V}_s} x_{i^t}^{f_k} = 1, \quad \forall f_k \in \mathcal{F}_r,\ t \in T, \tag{46}$$

$$x_{i^t}^{f_k} \in \{0, 1\}, \quad \forall i^t \in \mathcal{V}_s,\ f_k \in \mathcal{F}_r,\ t \in T, \tag{47}$$

$$y_{i^t j^k} \in \{0, 1\}, \quad \forall (i^t, j^k) \in \mathcal{E},\ t \in T, \tag{48}$$

where $\mathbf{x}' = \{x_{i^t}^{f_k} \in \{0,1\}, \forall i^t \in \mathcal{V}_s, f_k \in \mathcal{F}_r, t \in T\}$, and $\mathbf{y}' = \{y_{i^t j^k} \in \{0,1\}, \forall (i^t, j^k) \in \mathcal{E}, t \in T\}$. Binary variable $x_{i^t}^{f_k} \in \{0,1\}$ indicates whether function $f_k \in \mathcal{F}_r$ is orchestrated on $i^t \in \mathcal{V}_s$, and binary variable $y_{i^t j^k} \in \{0,1\}$ indicates whether the task passes $(i^t, j^k) \in \mathcal{E}$. The meanings of constraints (34)-(46) of PP are similar to the constraints in OP. PP is designed to find feasible service paths with the minimum reduced cost ζ_p,

$$\zeta_p = \sum_{t \in T} \sum_{(i^t, j^t) \in \mathcal{E}} \left(1 + \lambda_{i^t j^t}^{(30)}\right) \delta_r y_{i^t j^t} + \sum_{t \in T} \sum_{i^t \in \mathcal{V}_s} \left(\sum_{f_k \in \mathcal{F}_r} \lambda_{i^t}^{(31)} \delta_r \sigma_{f_k} x_{i^t}^{f_k} + \lambda_{i^t}^{(32)} E_{i^t}^{con,r} \right) - \lambda_r^{(29)}. \tag{49}$$

LRMP and PPs are solved iteratively until the termination condition is met. If $\zeta_p < 0$, the new generated service paths (new columns) from PPs are added to LRMP. If $\zeta_p \ge 0$, the new columns generated from PPs are no longer added to LRMP, and LRMP is solved using the existed columns. If the solution of LRMP is integer, it is the solution for RMP. Otherwise, if the solution of LRMP is not integer, the B&P algorithm is explored to obtain the optimal solution, detailed in Section IV-C. Note that each PP corresponding to a task is independent with each other due to the block angular structure. Hence, all PPs can be solved in parallel, which accelerates the solution procedure.

C. Branch-and-Price

To guarantee the optimal solution, we introduce the B&P algorithm to deal with the problem, as specified in Algorithm 1. Note that the traditional branch strategy based on the variables of RMP is not effective for B&P, since if a column is cut in LRMP when branch, it may be regenerated as a new column by PP and added to LRMP again, leading to the endless loop [12]. In order to avoid such problems, a branching rule compatible with PP should be adopted. To cope with this issue, we leverage the branch strategy on the original variable $x_{i^t}^{r,f_k}$ and $y_{i^t j^k}^r$. Since $x_{i^t}^{r,f_k} = \sum_{p \in \mathcal{P}_r} z_p^r u_{i^t}^{p,f_k}$ and $y_{i^t j^k}^r = \sum_{p \in \mathcal{P}_r} z_p^r w_{i^t j^k}^p$, it is equivalent to branch on $\sum_{p \in \mathcal{P}_r} z_p^r u_{i^t}^{p,f_k}$ and $\sum_{p \in \mathcal{P}_r} z_p^r w_{i^t j^k}^p$. It is significant to determine the upper bound (UB) since UB will finally converge to the optimal solution. At the very beginning, UB $\overline{\mathcal{Z}}$ is initialized by Algorithm 2 and lower bound (LB) is initialized as $\underline{\mathcal{Z}} = 0$ (line 2). Then, at each node of the search tree, CG is employed to update the UB and LB (line 5-22). Specifically, if the solution of LRMP $\mathcal{Z}_{LRMP}$ is integer and $\mathcal{Z}_{LRMP} > \overline{\mathcal{Z}}$, the current node and its corresponding subtree are pruned (line 13-14). If the solution of LRMP $\mathcal{Z}_{LRMP}$ is integer and less than the incumbent UB

Algorithm 1 B&P Algorithm

Input: SDTEG $\mathcal{G}(\mathcal{V}, \mathcal{E})$, number of time slots T, node computation capacity $\mathcal{C}(i^t)$, link communication capacity $c(i^t, j^t)$, tasks $r \in \mathcal{R}$ with $(o^t, d^{t'}, \delta_r, \mathcal{F}_r)$.

Output: $\mathbf{x}$, $\mathbf{y}$, and optimal solution $\mathcal{Z}^*$.

1: *// Initialization:*
2: Initialize the parameters as the root node of the search tree. Get an initial feasible solution as the upper bound $\overline{\mathcal{Z}}$ by Algorithm 2, and set the initial lower bound $\underline{\mathcal{Z}} = 0$.
3: **if** the search tree $\neq \emptyset$ **then**
4: *// Branch:*
5: Select a node in the searching tree.
6: Initialize the columns $\mathcal{P}_r'$ for the current RMP using Algorithm 2.
7: *// Column generation:*
8: Solve the LRMP of the current node to get dual variables $\boldsymbol{\lambda}$ and put $\boldsymbol{\lambda}$ into PP.
9: **repeat**
10: Solve PP using Algorithm 3 to get new columns and ζ_p. Add the new columns to current LRMP ($\mathcal{P}_r' = \mathcal{P}_r' \cup$ columns) and go to 8.
11: **until** $\zeta_p \ge 0$.
12: Solve the current LRMP to get the optimal value $\mathcal{Z}_{LRMP}$.
13: **if** $\mathcal{Z}_{LRMP}$ is integer and $\mathcal{Z}_{LRMP} > \overline{\mathcal{Z}}$ **then**
14: Prune the current node and its corresponding subtree.
15: **else if** $\mathcal{Z}_{LRMP}$ is integer and $\mathcal{Z}_{LRMP} \le \overline{\mathcal{Z}}$ **then**
16: Update incumbent $\overline{\mathcal{Z}} = \mathcal{Z}_{LRMP}$. Delete the current node and go to 4.
17: **else if** $\mathcal{Z}_{LRMP}$ is fractional **then**
18: Update incumbent $\underline{\mathcal{Z}} = \mathcal{Z}_{LRMP}$.
19: **if** $\underline{\mathcal{Z}} > \overline{\mathcal{Z}}$ **then**
20: Prune the current node and its corresponding subtree.
21: **else**
22: Delete the current node and add its child nodes in the search tree. Go to 4.
23: **end if**
24: **end if**
25: **end if**
26: $\mathcal{Z}^* = \overline{\mathcal{Z}}$.

Algorithm 2 Initialization Algorithm for RMP

1: Sort $\mathcal{R}$ according to δ_r in a descending order.
2: Remove the links in SDTEG that do not satisfy the bandwidth constraint in SDTEG.
3: **for** $r \leq |\mathcal{R}|$ **do**
4: Find K routing paths P_r^K for r in SDTEG leveraging the resource constrained K-shortest path algorithm.
5: **if** $P_r^K = \emptyset$ **then**
6: Task r is discarded.
7: **else**
8: Choose the shortest path in P_r^K, and orchestrate $\mathcal{F}_r$ on the nodes of the path with the maximum computing resource and enough energy.
9: **if** VNF orchestration is found **then**
10: Service path for r is obtained, and refresh the residual resources in SDTEG.
11: **else**
12: Delete the current shortest path in P_r^K.
13: **if** $P_r^K \neq \emptyset$ **then**
14: Go to 8.
15: **else**
16: Task r is discarded.
17: **end if**
18: **end if**
19: **end if**
20: **end for**
21: **Return** a set of feasible columns for RMP, i.e., $w^p_{i^t j^t}$ and $u^{p,f_k}_{i^t}$.

Algorithm 3 Algorithm for PP

1: Using the PSCA to get the K shortest paths.
2: **for** $k \leq K$ **do**
3: Orchestrate $\mathcal{F}_r$ on the nodes with enough computing resource and energy of the path.
4: **if** failed **then**
5: Delete the path.
6: **end if**
7: **end for**
8: Select the service path with the least communication resource cost.
9: **if** $\zeta_p < 0$ **then**
10: Go to 19.
11: **else**
12: Solve PP using B&B Algorithm.
13: **if** $\zeta_p < 0$ **then**
14: Go to 19.
15: **else**
16: Go to 20.
17: **end if**
18: **end if**
19: **Return** the generated columns.
20: **Return** $\emptyset$.

$\overline{\mathcal{Z}}$, UB is updated as $\overline{\mathcal{Z}} = \mathcal{Z}_{LRMP}$ and the current search node is fathomed (line 15-16). If the solution of LRMP $\mathcal{Z}_{LRMP}$ is fractional, it is served as the LB $\underline{\mathcal{Z}}$ (line 17-18). Moreover, if LB is larger than UB, the current search node and its subtree are pruned (line 19-20), otherwise, branch taking place in the current search node (line 22). In addition, choosing a good search node can accelerate the search efficiency, and we use the best-node-first-search to select the next search node [37]. Detailed in our problem, we choose the node with the largest LB $\underline{\mathcal{Z}}$ in the branch tree. When the search node is confirmed, the preprocessing can reduce the searching space: 1) removing the impossible links and nodes in SDTEG before the start time slot t and after the end time slot t' of task r. 2) According to constraints (11) and (16), if a couple of variables are set to 1, other corresponding variables are set to 0. 3) The existed columns which violate the branch constraint are deleted. 4) Inspired by [38], a linear programming relaxation of OP to obtain a general LB before branching accelerates the searching speed. Since if the general LB is larger than UB, CG is no longer activated to obtain a better LB, and the node as well as its subtree are deleted.

1) Initialization for RMP: At each node of the search tree, an initial feasible solution for RMP should be provided to obtain the dual variables for LRMP and help to update UB. The initial columns are a set of feasible service paths satisfying all the constraints in OP, and one service path for each task. To obtain an initial feasible solution of RMP as UB, Algorithm 2 is designed, based on the K-shortest path algorithm [39] in SDTEG. To take full advantage of various resources, in Algorithm 2, tasks are sorted according to the data size δ_r (line 1), since if the largest task uses the shortest path, communication resource consumption will be reduced. Then, the links in SDTEG which cannot satisfy the bandwidth constraint are removed (line 2). In addition, according to the branching constraints, if a link (i^t, j^t) is not on the searching path, i.e., $y^r_{i^t j^t} = 0$, the communication resource of (i^t, j^t) in SDTEG is set to 0 before the initialization for r. Then, we employ the resource constrained K-shortest path algorithm to obtain K physical paths P_r^K for each r, if P_r^K does not exist, the corresponding task cannot be satisfied and discarded (line 4-6). Since other than routing, VNFs need to be orchestrated, and one shortest path may violate the resource constraint of VNF orchestration. As such, considering K shortest paths contributes to find a feasible service path. In each iteration, the shortest path will be chosen to orchestrate VNFs until a feasible service path is found for task r or r is discarded (line 8-19). Note that when a service path is found for a task, the residual resources in SDTEG are refreshed (line 10). Finally, a set of feasible columns (feasible paths) are found for RMP (line 21).

2) Solution for PP: PP for each r is a small scale ILP problem and B&B can be directly used to obtain the optimal solution. However, it is still time consuming due to the huge number of PP, and the multiple iterations between LRMP and PP. Note that it is not necessary to obtain the optimal solution of PP in each iteration between LRMP and PP, so when a negative reduced cost is satisfied, the iteration in PP can stop before the optimal solution is found [40]. As such, Algorithm 3 is designed to obtain eligible columns by combining heuristic

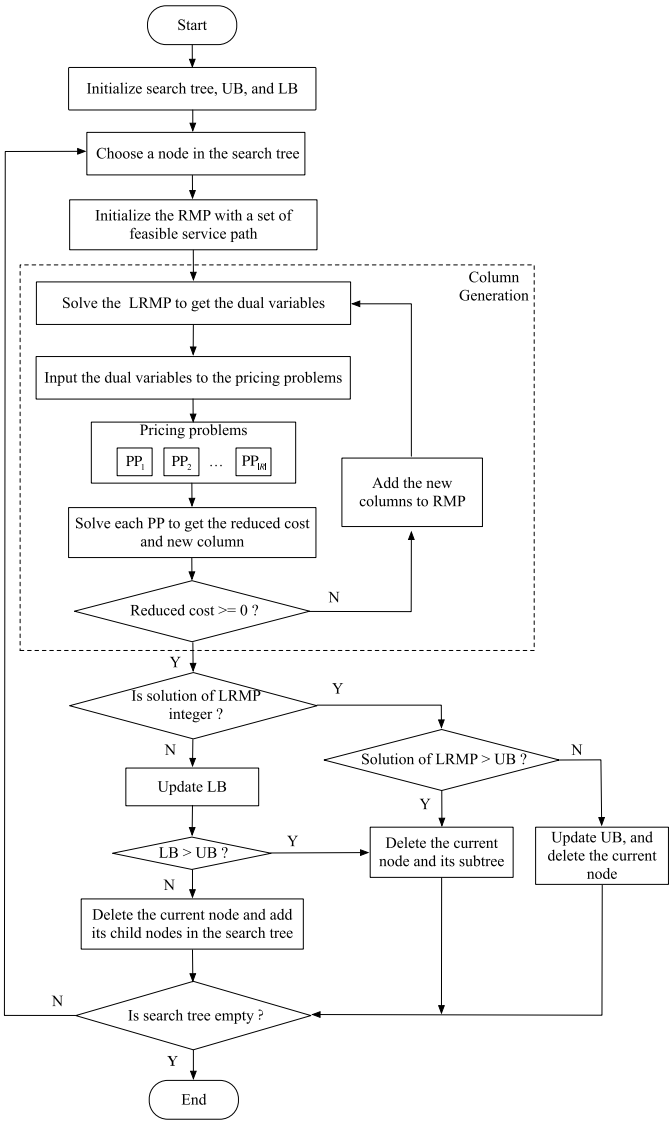


Fig. 3. Diagram of B&P scheme.

and B&B. In detail, referring to the path set construction algorithm (PSCA) in [23], the K shortest routing paths for r are constructed (line 1), and VNFs are orchestrated on the nodes of the constructed shortest paths to form service paths (line 3). If VNFs cannot be successfully orchestrated on the chosen path, the path is deleted (line 4-6). Then, the service path with the least communication resource consumption is selected (line 8). If it produces a negative reduced cost, the new generated column is added to LRMP (line 9-10). Otherwise, standard B&B algorithm is invoked to obtain an exact optimal solution for PP (line 12-14). If B&B also fails to find a new column with negative reduced cost, the treatment on the searching node is terminated (line 16).

Note that in the CG of each node in the search tree, LRMP is a small scale linear programming problem and can be directly solved by the optimization tool. PPs corresponding to different tasks are independent and can be solved in parallel. Each PP is still an ILP problem and in the worst case, the complexity

Algorithm 4 Approximation Algorithm for the Last RMP (ALR)

Input: Fractional solutions of LRMP with generated columns, and incumbent UB $\overline{\mathcal{Z}}$.
Output: $\mathbf{x}'$, $\mathbf{y}'$, and updated UB $\overline{\mathcal{Z}}$.
1: Sort $\mathcal{R}$ according to the task δ_r in a descending order.
2: **for** $r \leq |\mathcal{R}|$ **do**
3: Select the service path p with the highest value of z_p^r.
4: **if** $|p| > 2$ **then**
5: Select the service path with the least communication resource consumption.
6: **end if**
7: Round the corresponding z_p^r to 1.
8: **if** the rounded service path is infeasible **then**
9: Delete the service path, and go to 3.
10: **end if**
11: Set the other corresponding service paths $\mathcal{P}_r \backslash p$ as 0 according to constraint (23), and update the residual resources in SDTEG.
12: **end for**
13: **if** $\mathcal{Z}'_{RMP} > \overline{\mathcal{Z}}$ **then**
14: Delete the current searching node and its subtree.
15: **else**
16: Update $\overline{\mathcal{Z}} = \mathcal{Z}'_{RMP}$.
17: **end if**

of PP is $\mathcal{O}\left(N' \cdot M' \cdot 2^{N'}\right)$ [33], in which $N' = |T| \cdot |\mathcal{V}_s| \cdot (|\mathcal{F}| + |\mathcal{V}_s| + 3) - |\mathcal{V}_s|$ is the number of variables of PP, and $M' = |T| \cdot (2 + 3 \cdot |\mathcal{V}_s| + |\mathcal{V}_s| \cdot |\mathcal{F}| + |\mathcal{F}|) + |T| \cdot |\mathcal{V}_s| \cdot (4 + |\mathcal{V}_s|)$ is the number of constraints of PP. Although $\mathcal{O}\left(N' \cdot M' \cdot 2^{N'}\right)$ is exponentially related with N', compared with $\mathcal{O}\left(N \cdot M \cdot 2^{N}\right)$ of OP, both the coefficient $N' \cdot M'$ and exponent N' are in smaller scale. In practice, Algorithm 3 can help obtain the solution for PP more efficiently. Moreover, to clarify the relations of the designed algorithms, we provide a detailed diagram in Fig. 3. It is observed that the initialization for RMP (Algorithm 2), CG, and the solution for PP (Algorithm 3) are all combined in the framework of B&P.

D. Acceleration Methods for B&P

As above, B&P algorithm enables the optimal solution more efficient than standard B&B for OP. In practical application for large scale networks, a less elegant but much more efficient solution with a small optimal gap is applicable. To obtain a satisfied solution in an acceptable time, we leverage two tricks including the approximation algorithm for RMP and a beam search method for branch tree.

1) Approximation for RMP: To efficiently obtain a feasible UB, at the end of each CG iteration, if the LB from the last LRMP is not an integer and it is less than UB, an approximation method for the corresponding RMP is devised to obtain an integer solution. If the approximated integer solution is larger than current UB, the corresponding node and subtree are pruned. In this way, the search space is further decreased and we can obtain the final solution more quickly. If the approximated integer solution is less than current UB, the UB

TABLE III
MAJOR SIMULATION PARAMETERS

Parameter	Value	Parameter	Value	Parameter	Value
Number of satellites	16	$\mathcal{T}$	2h	τ	200s
Height of orbits	780km	P_o	5W	P_{ss}^{tr} ,P_{sd}^{tr}	30W
Constellation type	Walker	P_{ss}^{re},P_{os}^{re}	10W	E_c	10J
Number of orbits	4	θ	80%	δ_r	[100,1000]Mbit
Inclination	45deg	$c(i^t, j^t)$	[10,60]Mbps	n_f	2
Orbit period	6028s	σ_{f_k}	{5,10}unit/100Mbit	ε	0.01

is updated as the approximated integer solution. However, such a solution is not guaranteed optimal, but the optimal gap $\frac{|\overline{\mathcal{Z}}-\underline{\mathcal{Z}}|}{\underline{\mathcal{Z}}} < \varepsilon$ can be adjusted by the small value ε. In detail, inspired by the approximation algorithm in [26], we design an approximation algorithm for the last RMP and the specific procedure is detailed in Algorithm 4. Firstly, sort $\mathcal{R}$ according to the task δ_r in a descending order (line 1). Then, in each iteration, select the service path with the highest value of z_p^r (line 3). If the number of such service path is more than 1, the service path with a less communication resource consumption is further selected and rounded to 1 (line 4-7), since such a service path is likely to be the optimal selection. Then, the feasibility is verified since the resource capacity on the approximated service path may be violated (line 8-10). Moreover, according to constraint (23), set the other corresponding service paths $\mathcal{P}_r \backslash p$ as 0, and update the residual resources in SDTEG (line 11). Finally, if $\mathcal{Z}'_{RMP} > \overline{\mathcal{Z}}$, the current searching node and its subtree are cut, or update UB as $\overline{\mathcal{Z}} = \mathcal{Z}'_{RMP}$ (line 13-17). To embed Algorithm 4 into B&P, Algorithm 4 replaces line 18-23 of Algorithm 1.

2) Beam Search: To accelerate the pruning for the search tree of B&P, beam search [41] is introduced to reduce the search space. Instead of searching all the eligible nodes in the search tree, beam search only explores the promising nodes in the tree which is called the beam nodes. The number of beam nodes in each layer of the search tree is beam width and each layer can preset different beam width according to the LB of each node in the same layer. Since the search space is not complete, an optimal solution may potentially be pruned. However, if a solution satisfying the predefined gap ε is not found, the beam width can be widened until ε is satisfied. Both Algorithm 4 and the beam search method will accelerate pruning the search tree in B&P, and a suboptimal solution is obtained with reduced time complexity. There always exits a tradeoff between the optimal solution and time complexity in the ILP problem.

V. NUMERICAL RESULTS

In this section, we conduct a set of simulations using Matlab for the LEO satellite network, and the optimization tools CVX and GUROBI are also employed in the simulation. Specifically, the satellite network topology parameters generated from Satellite Tool Kit (STK) [42] are imported into Matlab. Matlab is employed for algorithm implementation, while CVX is used for solving LRMP, and GUROBI is used to obtain the optimal solution for PPs.

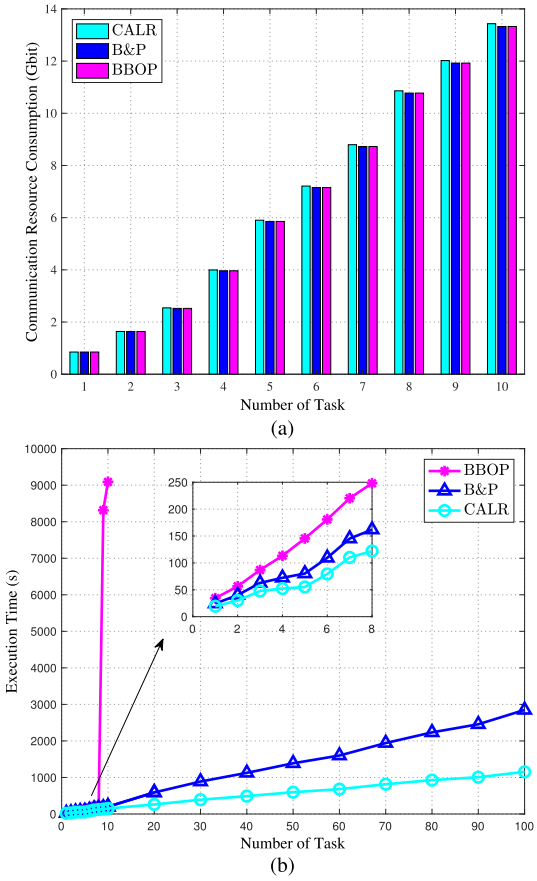


Fig. 4. Performance of B&P and CALR, with number of VNF enabled nodes 16, $\mathcal{D}(i^t)$ = 200units, and $EB_{i^t}^{max}$=30KJ. (a) Communication resource consumption *versus* number of task. (b) Execution time *versus* number of task.

A. Simulation Setup

The main parameters used in the simulation are listed in Table III. 16 satellites are uniformly distributed in 4 orbits at the height of 780km and inclination of 45 deg, and the orbit period is 6028s. Each orbit has 4 satellites, constituting a Walker constellation. Time horizon is set as 2 hours from 20 Jun 2019 06:00:00.000 UTCG to 20 Jun 2019 8:00:00.000 UTCG, with time slot length $\tau = 200$s. Besides, energy corresponding parameters are set as: $P_o = 5$W, $P_{ss}^{tr} = 20$W, $P_{sd}^{tr} = 20$W, $P_{ss}^{re} = 10$W, $P_{os}^{re} = 10$W, $E_c = 10$J, and $\theta = 80\%$ [30]. Ka frequency band is used for S2S. According to formulas (1)-(3), the S2S link capacity is obtained as [10,60] Mbps. The capacity of S2D and O2S is

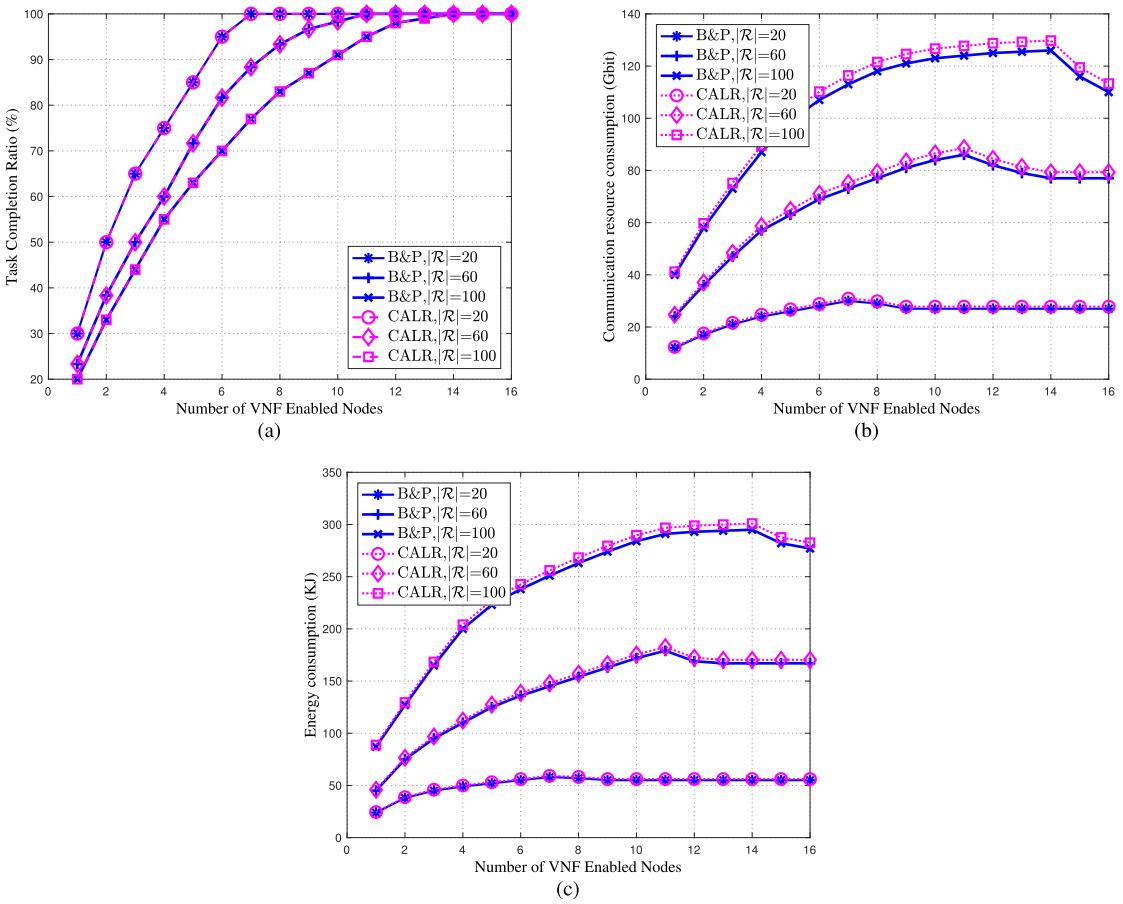


Fig. 5. Impact of VNF enabled nodes on network performance, with $\mathcal{D}(i^t) = 200$units and $EB_{i^t}^{max}$=30KJ. (a) Task completion ratio *versus* number of VNF enabled nodes. (b) Communication resource consumption *versus* number of VNF enabled nodes. (c) Energy consumption *versus* number of VNF enabled nodes.

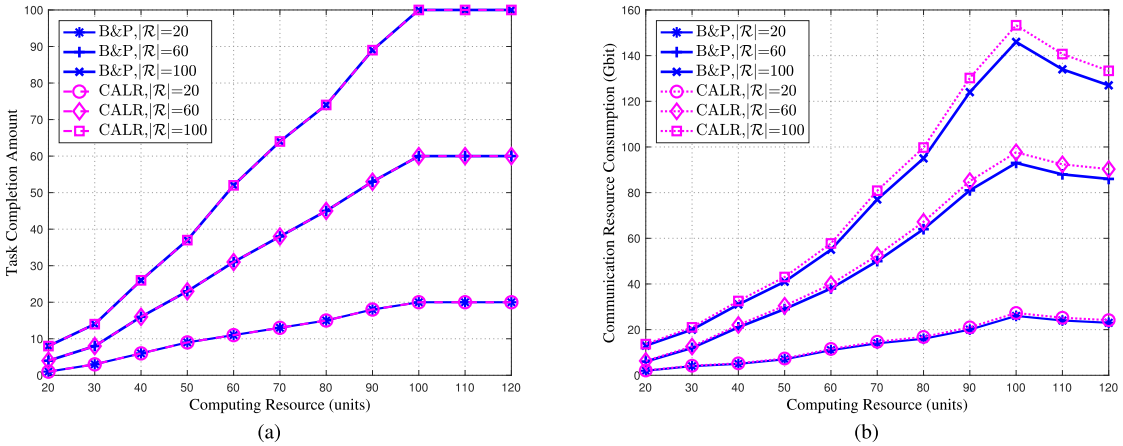
set as 100Mbps. We consider two types of VNF, i.e. $n_f = 2$, and the corresponding computing cost is set as 5unit/100Mbit and 10unit/100Mbit.

B. Performance Evaluation

To validate the optimality of B&P algorithm and performance of CALR (CG+ALR), we compare the proposed two methods with the optimal solution from BBOP (B&B for the original problem). Then, we explore the benefits from VNF orchestration. In addition, performance impacts from different network parameters such as computing capacity and battery capacity are also being evaluated. Note that ε is set as 0.01 in the following results.

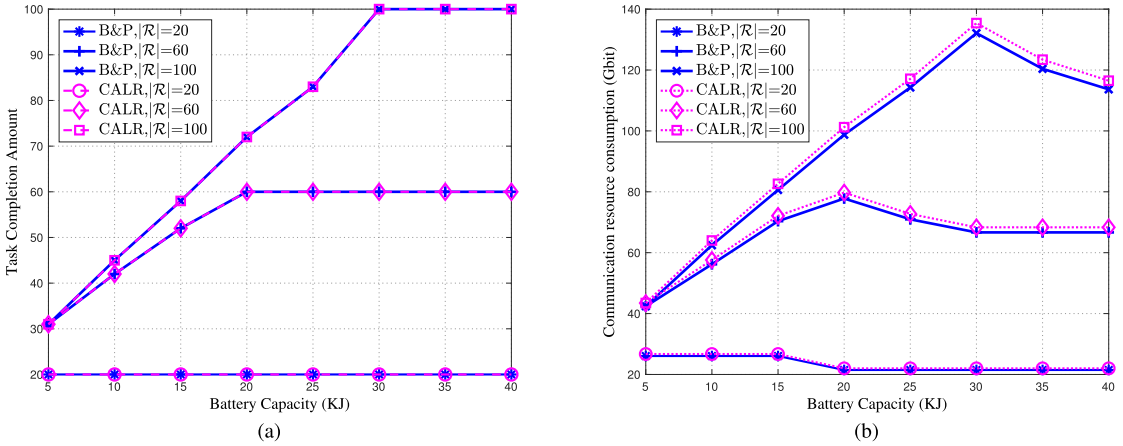
Firstly, the number of VNF enabled satellites is set as 16 and each satellite has 200 units of computing resource and battery capacity of 30KJ. By testing multi-group of 1-100 random tasks, Fig. 4 demonstrates the optimality of B&P and efficiency of CALR. In detail, "Communication Resource Consumption" in Fig. 4a is the optimization objective of the original formulated problem, which serves as a key metric to evaluate the effectiveness of the proposed algorithms. At the same time, the metric "Execution Time" in Fig. 4b is used to evaluate the time complexity for different algorithms execution. Compared with the optimal solution from BBOP in Fig. 4a, we observe that B&P can reach the same optimal solution. However, such comparisons are just suitable for small amount of tasks, since we cannot obtain the solution for OP from B&B searching due to the unacceptable solution time, for example, 10 tasks consume about 9000s, as shown in Fig. 4b. Thanks to the optimal property of B&P, we can employ its solution as the optimal benchmark. In regard to CALR, note that it can obtain a near optimal solution with high efficiency, and this advantage is outstanding with the increasing number of task. Another observation from Fig. 4b is that B&P and CALR have no superiority when the number of task is small, and the hidden reason is both B&P and CALR have iteration time cost between LRMP and PP, but this effect is neglected when there are a large amount of tasks in practice.

We further study the performance impacts from the number of VNF enabled LEO satellite node, which can provide insights into the NFV degree of SDLSN. In other words, the more satellites supporting VNF orchestration, SDLSN has a higher NFV degree. As shown in Fig. 5, the number of VNF enabled satellites changes from 1 to 16, and each satellite node is equipped with 200 units of computing resource and 30KJ of battery capacity. "Task Completion Ratio" in Fig. 5a is a key metric to evaluate the SDLSN performance with different VNF enabled LEO nodes, and in Fig. 5b, "Communication Resource Consumption" is the original optimization objective



(a) (b)

Fig. 6. Impact of computing resource on network performance, with number of VNF enabled nodes 16, and $EB_{it}^{max} = 30$KJ. (a) Task completion amount *versus* computing resource. (b) Communication resources consumption *versus* number of task.



(a) (b)

Fig. 7. Impact of battery capacity on network performance, with number of VNF enabled nodes 16, and $\mathcal{D}(i^t) = 200$units. (a) Task completion amount *versus* battery capacity. (b) Communication resources consumption *versus* battery capacity.

to investigate the proposed algorithms under different VNF enabled LEO nodes. In addition, since "Energy Consumption" is also a significant metric and in Fig. 5c, it is used to assess the proposed algorithms in the SDLSN with different VNF enabled LEO nodes. In Fig. 5a, considering $|\mathcal{R}| = 20$, $|\mathcal{R}| = 60$, and $|\mathcal{R}| = 100$ respectively, the task completion ratio gradually goes up with the increasing number of VNF enabled nodes until 100% completion. In detail, for $|\mathcal{R}| = 20$, the fewest number of VNF enabled nodes is 7 for 100% task completion, 10 for $|\mathcal{R}| = 60$, and 14 for $|\mathcal{R}| = 100$. This trend is expectable since if the number of VNF enabled nodes is small, some tasks are failed because there is no feasible physical path to the VNF enabled nodes or the path is limited by communication resources. Sufficient VNF enabled nodes provide more optional physical paths and increase the task success possibility. Fig. 5b depicts that with the increasing number of VNF enabled nodes, the amount of communication resource consumption firstly ascends and then goes down. For example, the inflexion point for $|\mathcal{R}| = 100$ is located at 14 VNF enabled nodes. Combined with the analysis of Fig. 5a, the reason for the ascending trend lies in that the increment of task completion brings new communication resource consumption until all the tasks are satisfied. In addition, there is a large amount communication resource consumption when there is a small number of VNF enabled nodes since a task should be transmitted many times to deploy the requested VNFs. This problem no longer exists when there are sufficient VNF enabled satellite nodes, since more flexible VNF orchestration guarantees lesser transmission times. As in Fig. 5c, we study the influence on the energy consumption from the number of VNF enabled nodes and it has a similar trend with the communication resource consumption in Fig. 5b. It can be further noted that, compared with Fig. 5b, after all the tasks are completed, the energy consumption decrease degree is less than communication resource consumption does. It can be explained that the energy cost by new completed tasks is composed of two parts: except for the energy cost by routing, the energy cost by VNF orchestration cannot be neglected.

To evaluate the effects of computing resource capacity of satellites, in Fig. 6, we study the task completion and communication resource consumption under different computing resource supply, i.e., $\{20, 30, \ldots, 120\}$ units. In detail, "Task

Completion Amount" in Fig. 6a evaluates the performance of proposed algorithms in SDLSN with different computing resource supply, and "Communication Resource Consumption" in Fig. 6b is the original optimization objective to investigate the proposed algorithms with varying computing resource capacity. Each satellite is equipped with the same amount of computing resource and with battery capacity of 30KJ. All satellites are VNF enabled. In Fig. 6a, some tasks fail when the computing resource is less than 100unit, and the trend is expectable since the number of completion task is limited by the VNF orchestration, which is further restricted by the computing resource. For example, when the computing resource is 50unit, it is a bottleneck for the task with data size more than 500Mbit. In Fig. 6b, with the increment of computing resource, the communication resource consumption firstly increases until the computing resource 100unit, and then goes down. It is accounted by the fact that the completion task number increases firstly and the upward trend stops when all the tasks are satisfied. The descending trend lies in that sufficient computing resources guarantee more opportunities for VNF orchestration, which can benefit the physical path length, i.e. reducing communication resource consumption. It is worth note that computing resource of 100unit is a critical point, since the orchestration for VNF with $\sigma_{f_k} = 0.1$ and data size of 1000Mbit requires 100unit in our problem.

In Fig. 7, we investigate the impact of battery capacity $\{5, 10, \ldots, 40\}$KJ on task completion performance and communication cost, i.e. "Task Completion Amount" in Fig. 7a and "Communication Resource Consumption" in Fig. 7b, which are key metrics to evaluate the proposed algorithms for SDLSN, with 16 VNF enabled nodes and each node with computing resource of 200units. As expected, in Fig. 7a, the number of task completion has an ascending trend. The communication consumption in Fig. 7b increases firstly and then descends, attributing to that increasing energy supply enables the network to accommodate more tasks, and each task can choose its preferred service path when the energy supply is abundant, which benefits the communication resource consumption.

VI. Conclusion

In this paper, we have explored the VNF orchestration based service provision in SDLSN, considering the intermittent S2S channel condition, energy capacity limitation and the limited computing payload. Since the corresponding formulated problem is intractable for large scale networks, we have designed the B&P algorithm by employing the DWD and CG methods to obtain the optimal solution efficiently. Moreover, in order to obtain effective solutions for practical applications, we have further proposed an approximation algorithm for RMP and employ the beam search method to accelerate the search tree pruning for B&P. Extensive simulation experiments have been conducted to verify that B&P can guarantee the optimal solution with less time cost, and the approximation methods can dramatically accelerate the efficiency of B&P with a small optimal gap. The impacts of different system parameters have been analyzed in the simulation, which will promote the resource management model design for SDLSN.

References

[1] Z. Jia, M. Sheng, J. Li, Y. Zhu, W. Bai, and Z. Han, "Virtual network functions orchestration in software defined LEO small satellite networks," in *Proc. ICC-IEEE Int. Conf. Commun. (ICC)*, Dublin, Ireland, Jun. 2020, pp. 1–6.

[2] W. Saad, M. Bennis, and M. Chen, "A vision of 6G wireless systems: Applications, trends, technologies, and open research problems," *IEEE Netw.*, vol. 34, no. 3, pp. 134–142, May 2020.

[3] Z. Jia, M. Sheng, J. Li, D. Zhou, and Z. Han, "Joint HAP access and LEO satellite backhaul in 6G: Matching game-based approaches," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 4, pp. 1147–1159, Apr. 2021.

[4] X. Zhang, J. Wang, C. Jiang, C. Yan, Y. Ren, and L. Hanzo, "Robust beamforming for multibeam satellite communication in the face of phase perturbations," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 3043–3047, Mar. 2019.

[5] J. Du, C. Jiang, H. Zhang, Y. Ren, and M. Guizani, "Auction design and analysis for SDN-based traffic offloading in hybrid satellite-terrestrial networks," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 10, pp. 2202–2217, Oct. 2018.

[6] *Spacex Plans to Put More Than 40,000 Satellites in Space*. Accessed: Oct. 2019. [Online]. Available: https://www.starlink.com/

[7] M. Sheng, Y. Wang, J. Li, R. Liu, D. Zhou, and L. He, "Toward a flexible and reconfigurable broadband satellite network: Resource management architecture and strategies," *IEEE Wireless Commun.*, vol. 24, no. 4, pp. 127–133, Aug. 2017.

[8] J. Liu, Y. Shi, L. Zhao, Y. Cao, W. Sun, and N. Kato, "Joint placement of controllers and gateways in SDN-enabled 5G-satellite integrated network," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 2, pp. 221–232, Feb. 2018.

[9] V. V. Vazirani, *Approximation Algorithms*. Berlin, Germany: Springer, Mar. 2013.

[10] J. Desrosiers and M. E. Lübbecke, *A Primer in Column Generation*. Boston, MA, USA: Springer US, 2005.

[11] F. Vanderbeck, "On Dantzig-Wolfe decomposition in integer programming and ways to perform branching in a branch- and-price algorithm," *Oper. Res.*, vol. 48, no. 1, pp. 111–128, Feb. 2000.

[12] C. Barnhart, E. L. Johnson, G. L. Nemhauser, M. W. P. Savelsbergh, and P. H. Vance, "Branch- and-price: Column generation for solving huge integer programs," *Oper. Res.*, vol. 46, no. 3, pp. 316–329, May 1998.

[13] G. Araniti, G. Genovese, A. Iera, A. Molinaro, and S. Pizzi, "Virtualizing nanosatellites in SDN/NFV enabled ground segments to enhance service orchestration," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Waikoloa, HI, USA, Dec. 2019, pp. 1–6.

[14] A. Papa, T. D. Cola, P. Vizarreta, M. He, C. Mas-Machuca, and W. Kellerer, "Design and evaluation of reconfigurable SDN LEO constellations," *IEEE Trans. Netw. Service Manage.*, vol. 17, no. 3, pp. 1432–1445, Sep. 2020.

[15] F. Tang, "Dynamically adaptive cooperation transmission among satellite-ground integrated networks," in *Proc. IEEE INFOCOM-IEEE Conf. Comput. Commun.*, Toronto, ON, Canada, Jul. 2020, pp. 1559–1568.

[16] D. K. Luong, Y.-F. Hu, J.-P. Li, and M. Ali, "Metaheuristic approaches to the joint controller and gateway placement in 5G-satellite SDN networks," in *Proc. ICC-IEEE Int. Conf. Commun. (ICC)*, Dublin, Ireland, Jun. 2020, pp. 1–6.

[17] T. Li, H. Zhou, H. Luo, and S. Yu, "SERvICE: A software defined framework for integrated space-terrestrial satellite communication," *IEEE Trans. Mobile Comput.*, vol. 17, no. 3, pp. 703–716, Mar. 2018.

[18] N. Font, C. Blosse, P. Lautier, A. Barthère, and P. Voisin, "Flexible payloads for telecommunication satellites–a Thales perspective," in *Proc. 32nd AIAA Int. Commun. Satell. Syst. Conf.*, San Diego, CA, USA, Aug. 2014, p. 4382.

[19] *China's 34th Launch of 2018 Places Five Satellites in Orbit*. Accessed: Nov. 2018. [Online]. Available: https://space.skyrocket.de/doc_sdat/tianzhi-1.htm

[20] *Chameleon Satellite to Revolutionise Telecom Market*. Accessed: Jul. 2015. [Online]. Available: https://space.skyrocket.de/doc_sdat/eutelsat-quantum.htm

[21] *Lockheed Martin's First Smart Satellites are Tiny With Big Missions*. Accessed: Mar. 2019. [Online]. Available: https://news.lockheedmartin.com

[22] X. Jia, T. Lv, F. He, and H. Huang, "Collaborative data downloading by using inter-satellite links in LEO satellite networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 3, pp. 1523–1532, Mar. 2017.

[23] S. El Alaoui and B. Ramamurthy, "Routing optimization for DTN-based space networks using a temporal graph model," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Kuala Lumpur, Malaysia, May 2016, pp. 1–6.

[24] M. Hussein, G. Jakllari, and B. Paillassa, "On routing for extending satellite service life in LEO satellite networks," in *Proc. IEEE Global Commun. Conf.*, Austin, TX, USA, Dec. 2014, pp. 2832–2837.
[25] Y. Yang, M. Xu, D. Wang, and Y. Wang, "Towards energy-efficient routing in satellite networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3869–3886, Dec. 2016.
[26] A. Jarray and A. Karmouch, "Decomposition approaches for virtual network embedding with one-shot node and link mapping," *IEEE/ACM Trans. Netw.*, vol. 23, no. 3, pp. 1012–1025, Jun. 2015.
[27] N. Huin, A. Tomassilli, F. Giroire, and B. Jaumard, "Energy-efficient service function chain provisioning," *IEEE J. Opt. Commun. Netw.*, vol. 10, no. 3, pp. 114–124, Mar. 2018.
[28] 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, Sep. 2017.
[29] Z. Jia *et al.*, "Joint optimization of VNF deployment and routing in software defined satellite networks," in *Proc. IEEE 88th Veh. Technol. Conf. (VTC-Fall)*, Aug. 2018, pp. 1–5.
[30] A. Golkar and I. L. I. Cruz, "The federated satellite systems paradigm: Concept and business case evaluation," *Acta Astronautica*, vol. 111, pp. 230–248, Jun. 2015.
[31] J. N. Pelton, S. Madry, and S. Camacho-Lara, *Handbook of Satellite Applications*. New York, NY, USA: Springer-Verlag, 2013.
[32] D. Zhou, M. Sheng, B. Li, J. Li, and Z. Han, "Distributionally robust planning for data delivery in distributed satellite cluster network," *IEEE Trans. Wireless Commun.*, vol. 18, no. 7, pp. 3642–3657, Jul. 2019.
[33] G. Nemhauser and L. Wolsey, *Computational Complexity*. Hoboken, NJ, USA: Wiley, 2014, ch. 1.5, pp. 114–145.
[34] G. B. Dantzig and M. N. Thapa, *Linear Programming 2: Theory and Extensions*. New York, NY, USA: Springer-Verlag, 2003.
[35] R. K. Martin, "Generating alternative mixed-integer programming models using variable redefinition," *Oper. Res.*, vol. 35, no. 6, pp. 820–831, Dec. 1987.
[36] F. Vanderbeck and L. A. Wolsey, "An exact algorithm for IP column generation," *Oper. Res. Lett.*, vol. 19, no. 4, pp. 151–159, Oct. 1996.
[37] G. C. Menezes, G. R. Mateus, and M. G. Ravetti, "A branch and price algorithm to solve the integrated production planning and scheduling in bulk ports," *Eur. J. Oper. Res.*, vol. 258, no. 3, pp. 926–937, May 2017.
[38] T. Nishi and T. Izuno, "Column generation heuristics for ship routing and scheduling problems in crude oil transportation with split deliveries," *Comput. Chem. Eng.*, vol. 60, pp. 329–338, Jan. 2014.
[39] D. Eppstein, "Finding the K shortest paths," *SIAM J. Comput.*, vol. 28, no. 2, pp. 652–673, 1998.
[40] G. Desaulniers, "Branch- and-price- and-cut for the split-delivery vehicle routing problem with time windows," *Oper. Res.*, vol. 58, no. 1, pp. 179–192, Feb. 2010.
[41] P. Slow and T. E. Morton, "Filtered beam search in scheduling," *Int. J. Prod. Res.*, vol. 26, no. 1, pp. 35–62, Jan. 1988.
[42] X. Zhang *et al.*, "Hardware-in-the-loop simulation system for space information networks," *J. Commun. Inf. Netw.*, vol. 2, no. 4, pp. 131–141, Dec. 2017.

Ziye Jia received the B.E. degree in communications engineering and the M.S. degree in electronics and communications engineering from Xidian University, Xi'an, China, in 2012 and 2015, respectively, where she is currently pursuing the Ph.D. degree in communications and information systems. She was a Visiting Ph.D. Student with the Department of Electrical and Computer Engineering, University of Houston, from 2018 to 2020. Her current research interests include resource allocation and NFV techniques for space-air-ground networks.

Min Sheng (Senior Member, IEEE) joined Xidian University in 2000, where she is currently a Full Professor and the Director of the State Key Laboratory of Integrated Services Networks. She has published over 200 refereed articles in international leading journals and key conferences in the area of wireless communications and networking. Her current research interests include space-terrestrial integration networks, intelligent wireless networks, and mobile ad hoc networks. She received the China National Funds for Distinguished Young Scientists in 2018. She is also the Vice Chair of IEEE Xi'an Section. She is also an Editor of IEEE COMMUNICATIONS LETTERS and IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS.

Jiandong Li (Fellow, IEEE) received the B.E., M.S., and Ph.D. degrees in communications engineering from Xidian University, Xi'an, China, in 1982, 1985, and 1991, respectively. He has been a Faculty Member of the School of Telecommunications Engineering, Xidian University, since 1985, where he is currently a Professor and the Vice Director of Academic Committee of the State Key Laboratory of Integrated Service Networks. He was a Visiting Professor with the Department of Electrical and Computer Engineering, Cornell University, from 2002 to 2003. His research interests include wireless mobile communication, cognitive, and self-organizing networks. He was awarded as a Distinguished Young Researcher from NSFC and a Changjiang Scholar from Ministry of Education, China, respectively. He has served as the General Vice Chair for ChinaCom 2009 and a TPC Chair for IEEE ICCC 2013.

Di Zhou (Member, IEEE) received the B.E. and Ph.D. degrees in communication and information systems from Xidian University, Xi'an, China, in 2013 and 2019, respectively. She was a Visiting Ph.D. Student with the Department of Electrical and Computer Engineering, University of Houston, from 2017 to 2018. Since 2019, she has been with the Broadband Wireless Communications Laboratory, School of Telecommunications Engineering, Xidian University, where she currently holds a faculty post-doctoral position. Her research interests include routing, resource allocation, and mission planning in space information networks.

Zhu Han (Fellow, IEEE) received the B.S. degree in electronic engineering from Tsinghua University, in 1997, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Maryland, College Park, MD, USA, in 1999 and 2003, respectively. From 2000 to 2002, he was a Research and Development Engineer with JDSU, Germantown, MD, USA. From 2003 to 2006, he was a Research Associate with the University of Maryland. From 2006 to 2008, he was an Assistant Professor with Boise State University, Idaho. He is currently a John and Rebecca Moores Professor with the Electrical and Computer Engineering Department, University of Houston, Houston, TX, USA, and the Computer Science Department, University of Houston. His research interests include wireless resource allocation and management, wireless communications and networking, game theory, big data analysis, security, and smart grid. He was/has been an IEEE Communications Society Distinguished Lecturer from 2015 to 2018, the AAAS fellow since 2019, and an ACM distinguished Member since 2019. He has been 1% highly cited researcher since 2017 according to Web of Science. He is also the winner of 2021 IEEE Kiyo Tomiyasu Award, for outstanding early to mid-career contributions to technologies holding the promise of innovative applications, with the following citation: "for contributions to game theory and distributed management of autonomous communication networks. He received an NSF Career Award in 2010, the Fred W. Ellersick Prize of the IEEE Communication Society in 2011, the EURASIP Best Paper Award for the *Journal on Advances in Signal Processing* in 2015, IEEE Leonard G. Abraham Prize in the field of Communications Systems [Best Paper Award in IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS (JSAC)] in 2016, and several best paper awards in IEEE conferences.